

UNCLASSIFIED

AD NUMBER

**AD488327**

NEW LIMITATION CHANGE

TO

**Approved for public release, distribution  
unlimited**

FROM

**Distribution authorized to U.S. Gov't.  
agencies and their contractors;  
Administrative/Operational Use; May 1966.  
Other requests shall be referred to the  
U.S. Naval Postgraduate School, Monterey,  
CA.**

AUTHORITY

**USNPS ltr, 21 Apr 1972**

THIS PAGE IS UNCLASSIFIED

488327

UNITED STATES  
NAVAL POSTGRADUATE SCHOOL



THESIS

AN ON-LINE SIMULATION OF ASW IN A  
MULTI-BURST NUCLEAR ENVIRONMENT

by

David Lee McMichael

May 1966

This document is subject to special export controls and each transmittal to foreign government or foreign nationals may be made only with prior approval of the U. S. Naval Postgraduate School.

Reproduced From  
Best Available Copy

AN ON-LINE SIMULATION OF ASW IN A  
MULTI-BURST NUCLEAR ENVIRONMENT

by

David L. McMichael  
Lieutenant, United States Navy  
B. S., Tri-State College, 1959

Submitted in partial fulfillment  
for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

UNITED STATES NAVAL POSTGRADUATE SCHOOL

May 1966

Signature of Author

*David L. McMichael*  
Operations Analysis Curriculum, May, 1966

Certified by:

*Mitchell H. Cotton*

Thesis Advisor

Accepted by:

*J. R. Brinkley*  
Chairman, Department of Operations Analysis

Approved by:

*R. J. Ruehart*

Academic Dean

Reproduced From  
Best Available Copy

## **REPRODUCTION QUALITY NOTICE**

This document is the best quality available. The copy furnished to DTIC contained pages that may have the following quality problems:

- Pages smaller or larger than normal.
- Pages with background color or light colored printing.
- Pages with small type or poor printing; and or
- Pages with continuous tone material or color photographs.

Due to various output media available these conditions may or may not cause poor legibility in the microfiche or hardcopy output you receive.



If this block is checked, the copy furnished to DTIC contained pages with color printing, that when reproduced in Black and White, may change detail of the original copy.

#### ABSTRACT

A general approach is documented as a guide to aid in the formulation and implementation of on-line, real time computer simulations. A computer program MULNUC1, is developed as an on-line, real time computer simulation of antisubmarine warfare in a multiple burst nuclear environment. The principals of the game are a submarine armed with torpedoes, and two destroyers equipped with stand-off antisubmarine weapons. The simulation is intended as a demonstration of the on-line capabilities of the United States Naval Postgraduate School computer system and as a tool for further study of the factors involved in a representative ASW operational environment.

## TABLE OF CONTENTS

Section	Page
1. Introduction	7
2. Formulation of On-Line Simulations	9
Initializing	9
An Iterative Loop	10
A critique	11
3. Program Modularization	12
Advantages	12
Flexibility	13
4. System Variables, Coordinates and Notation	14
5. MULNUC1 - An Example	16
Executive Control	18
Plot Generator	21
Interactions	24
Sonar Contact Model	24
Contact Tracking Model	25
Weapon Firing Model	25
Evaluation Model	26
Radiation Model	26
Submarine Logic Model	27
Sequencing of Blocks	28
Man Machine Interface	30
Timing	36
6. Conclusions and Acknowledgements	38
Bibliography	39

Section	Page
Appendix I - List of Variables	40
Appendix II - Logic Flow Diagrams	50
Appendix III - Subroutines and CDC 160 Executive Routine	69
Appendix IV - Program Listing	74

## LIST OF ILLUSTRATIONS

Figure	Page
1. Schematic Layout of Computer Complex	17
2. Typical Left Tube Display on DD 65	22
3. Typical Right Tube Display on DD 65	23
4. DD 65 Console Arrangement	33
5. Overlay used on DD 65 Keyboard	35

## 1. INTRODUCTION

Simulation is a useful tool of the operations analyst. This is not a new concept, the first recorded simulations were conducted by the Chinese in the form of "war chess" and were probably used to teach young men some of the concepts of battle without the inherent danger of loss of life. Later accounts of simulations have been recorded by the Prussians, French and Germans [7].

This technique of analyzing a problem by simulation is now employed by all branches of the scientific community. The present day high speed digital computer has given rise to a rapid expansion in the use of simulation as a method of solution to military, scientific, management and many other types of problems.

Convincing the reader or observer that a simulation "models the real world" is one of the primary problems confronting the analyst who uses simulation techniques in the solution of a problem. One way to minimize this doubt is to increase the role of the human in the simulation. This can, in many cases, be done by the techniques of on-line simulation. While on the one hand, on-line simulation increases the complexity of the problem by nature of the man machine interface, at the same time, on-line simulation adds very complex logic (the man) to the problem with minimal effort on the part of the designer.

The design of a simulation, and in particular an on-line simulation, can appear to be a formidable task when first considered. However, if the designer has an approach in mind and proceeds in an organized manner, the problem usually divides into small parts that are each relatively simple. It is the aim of this thesis to take a representative

problem and develop a computer simulation that can be used as a reference for the construction of on-line, real time, computer simulations in general.

## 2. FORMULATION OF ON-LINE SIMULATIONS

In the formulation of on-line simulations the designer must first lay down a good foundation in the form of a well-planned outline. This outline must begin by initializing and setting the scene for the simulation. When this has been accomplished the designer must turn his attention to the development of a loop that will include all the actions and interactions expected to occur in the simulation. Included as an integral part of this loop is a timing mechanism that is flexible enough to allow the simulation to proceed at any rate required. A third part of the formulation is concerned with providing a critique of the simulation, either as a running critique or a compilation of pertinent facts at the end of the simulation.

### INITIALIZING

Initializing is the term applied to that portion of the simulation which is executed before "play" begins. It includes data input, assignment of particular values to the parameters and entering starting values required for indexing the logic. Herein is provided the flexibility required of any useful simulation. In the initializing portion the ground work must be formed for performing sensitivity analysis if such analysis is required. The initializing portion of the simulation must allow enough flexibility to provide for the various scenarios possible in the particular simulation. Clearly then, the initializing portion must be designed with these purposes as primary criteris and subject to boundary conditions, such as equipment capabilities.

## AN ITERATIVE LOOP

In general, many on-line simulations contain an iterative loop that is cycled for each time period. Therefore, one of the first considerations to be made in this phase of the design is the determination of the stepping interval. Several factors are involved in this selection, the most important being the assurance that the stepping interval is compatible with the logic flow of the situation being simulated. The designer must then consider the amount of time required to accomplish the most complicated simulation situation that can occur in one cycle. These considerations complete, the simulator must insure that the cycling is such that the player is not bored with the data/action as presented and also that this data/action is not presented at a rate too rapid for the player to fully comprehend. With these considerations in mind, a tentative looping cycle can be constructed and the designer may continue to develop the necessary routines required to complete the iterative loop.

The loop should now be fashioned in its most elementary form. The designer must consider several tasks that must be accomplished during each cycle. These include, for example, advancing all participants one time cycle, consideration of the possible interactions that can occur due to these moves, tabulation of the results of such interactions, presentation of output to the player, permitting the player to communicate with the simulation, delaying the next cycle until the proper time interval has transpired, and possible other considerations dependent upon the particular simulation.

Care and planning must be exercised in the construction of this loop since this is the foundation upon which the designer is to build his simulation. If logical errors appear in the order of these routines or

a component is not considered in this loop, the remaining portion of the design will be difficult, if not impossible. Planning in this portion of the development will be time well spent.

#### THE CRITIQUE

The critique is that portion of the simulation in which the entire simulation, or any integral part, is analyzed and the results compiled in condensed form. There are several basic techniques that may be explored in this part of the simulation. The least complex of which is, in most cases, a complete "recording" of the game that can be "replayed" at a later time at any speed desired. A more complex approach to the problem of performing a critique of a simulation is that of including a recording/analyzing routine in the iterative loop. This routine would extract the desired information during each cycle of the loop. It is apparent that not every cycle need be recorded, and therefore, a decision logic must be included that will extract all necessary information. Associated with this technique must be a recording routine that can be queried at intervals or at the conclusion of the simulation.

Much care must be exercised in formulating the design and location of the critique routine since this is the major mode of analysis available to the analyst. Flexibility and adaptability are considerations that must be made to allow the simulation to be fully appreciated as an analytical tool and not just a "parlor game."

### **3. PROGRAM MODULARIZATION**

In general, programmers attempt to modularize their programs. There are several reasons for this, the most obvious being that of providing logical grouping of ideas. The large scale simulation is usually modularized by the use of subroutines. This use of subroutines is convenient because:

a. Several separate groups may be working on various sections of the problem, and in many instances the use of subroutines is the best technique.

b. Computations which are to be called upon several times in the main program are best handled by the use of subroutines.

c. The program may be of a magnitude such that the entire program cannot be compiled in one pass.

These reasons are valid for the large scale simulation. In simulations which are moderate to small in size the use of subroutines may add unnecessary factors to be considered, with the exception of reason (b) above which is a valid reason for the use of subroutines in most computer applications.

By careful construction of the statement numbering scheme, available in languages such as FORTRAN, the programmer of the moderate to small size simulation can modularize his program without the additional consideration of designating common storage and the other difficulties experienced when programming subroutines.

#### **ADVANTAGES**

The above technique allows the designer to use the familiar computer languages, such as FORTRAN, in place of special simulation languages.

It also alleviates the requirement of either providing the reader with a description of a special language or causing him to go to another source to interpret the program. Using a "standard" language, such as FORTRAN, the designer can reasonably assume that the reader needs little or no explanation.

#### **FLEXIBILITY**

The use of modularization with statement numbers gives all the inherent flexibility observed in the special languages when the simulation is of such a magnitude as to allow compilation in one pass. When simulations are in the design and programming stages, the use of the above technique can allow as much flexibility as the simulation requires. Exit from the blocks when programming in FORTRAN can be accomplished at any logical point with nothing more than a simple GO TO or COMPUTED GO TO statement, and thus the logic flow is easily accomplished by this method.

#### 4. SYSTEM VARIABLES, COORDINATES AND NOTATION

Certain special simulation languages provide for dynamic allocation of storage for tables which facilitates the designation of data in a flexible and expandable form. This is not necessarily required by the moderate size simulation. The designer of these moderate size simulations may find the techniques as explained below more desirable, since his problem is not generally one of storage limitations. The criterion for the dynamic versus preset storage decision is felt to be that of program size. In programming the moderate size simulation the programmer may find that constructing a scheme for naming variables may create easier to work with variables than the complex tabular form of the special simulation language. This was found to be true in the programming of the example simulation, MULNUC1, of this thesis.

The general approach to the naming of variables in the example program was that of using vectors to represent a given parameter, each component of the vector being representative of the value of that parameter with respect to the unit concerned. An example, that of the X-coordinate of the  $i^{\text{th}}$  destroyer, being DDX(I). Once the schema is understood the programming moves along without a great amount of thought required by the programmer as far as variable names are concerned. Certain variables must be set aside as dummy or temporary and these logically take on forms such as: ITEMP, TEMP3, DUMMY(I), and forms that immediately classify them in this category.

In general, several different coordinate systems are required in simulations. In the war game an overall "area of play" must be established. This can be either rectangular or polar, two or three dimensional. The

axes of the rectangular coordinate system are not necessarily graduated in the same units. In the example program, for instance, it will be seen that the third axis (depth) is dimensioned in feet, while the two major axes are dimensioned in yards. Further, the third axis has the normally negative direction established as positive. These, perhaps unorthodox, measures are taken to facilitate programming; however, they must be spelled out in the documentation of the simulation to prevent possible misunderstanding.

Many simulations require more than one coordinate system to be employed. The overall play is perhaps in a rectangular coordinate system, while range and bearing information may be required during the play of the game. This will necessitate the incorporation of an overlay of one, or perhaps several, polar coordinate systems upon the base system. If a "close-up" view is required during the play a translation and/or expansion to another rectangular system may be required. It can now be seen that, in general, several coordinate systems will be used in a war game type of simulation. Therefore, a plan for the designation of these various coordinate systems and their respective transformations must be established early in the formulation of a simulation.

### 5. MULNUC1 - AN EXAMPLE

MULNUC1 is an on-line, real time simulation used as an example of the application of ideas presented in sections one through four. The simulation had its beginning at the Naval Radiological Defense Laboratory, Hunters Point, in the summer of 1965. During a six-week tour of the Naval Radiological Defense Laboratory, it was found that little had been done in the exploration of tactics and possible reactions of surface antisubmarine destroyers exposed to a self-inflicted multiple burst nuclear environment. At this time the simulation used as an example in this thesis had its beginning. In the example program, MULNUC1, all classified input parameters have been assigned fictitious values so that the computer program, as presented in this thesis, could remain unclassified.

In 1965, Lieutenant J. E. Johnson programmed the on-line display, Display Data Corporation model DD 65, using a rather simple simulation situation [2]. He did make a contribution in the form of an advancement in the techniques of on-line programming of simulations. Having observed a demonstration of Johnson's program, it was felt that the technique of on-line display would be ideal for the envisioned program, MULNUC1.

Several links were missing in the chain necessary to put the envisioned program on-line. The first link was a requirement for a routine to generate circles of arbitrary size and location. This was accomplished with subroutine CIRCLE (see Appendix III). After completing subroutine CIRCLE, attention was turned to the necessity for a random number generator capable of generating several types of random variables. The distributions of random variables required were uniform, normal, and circular normal. These generators were written in the form of subroutines

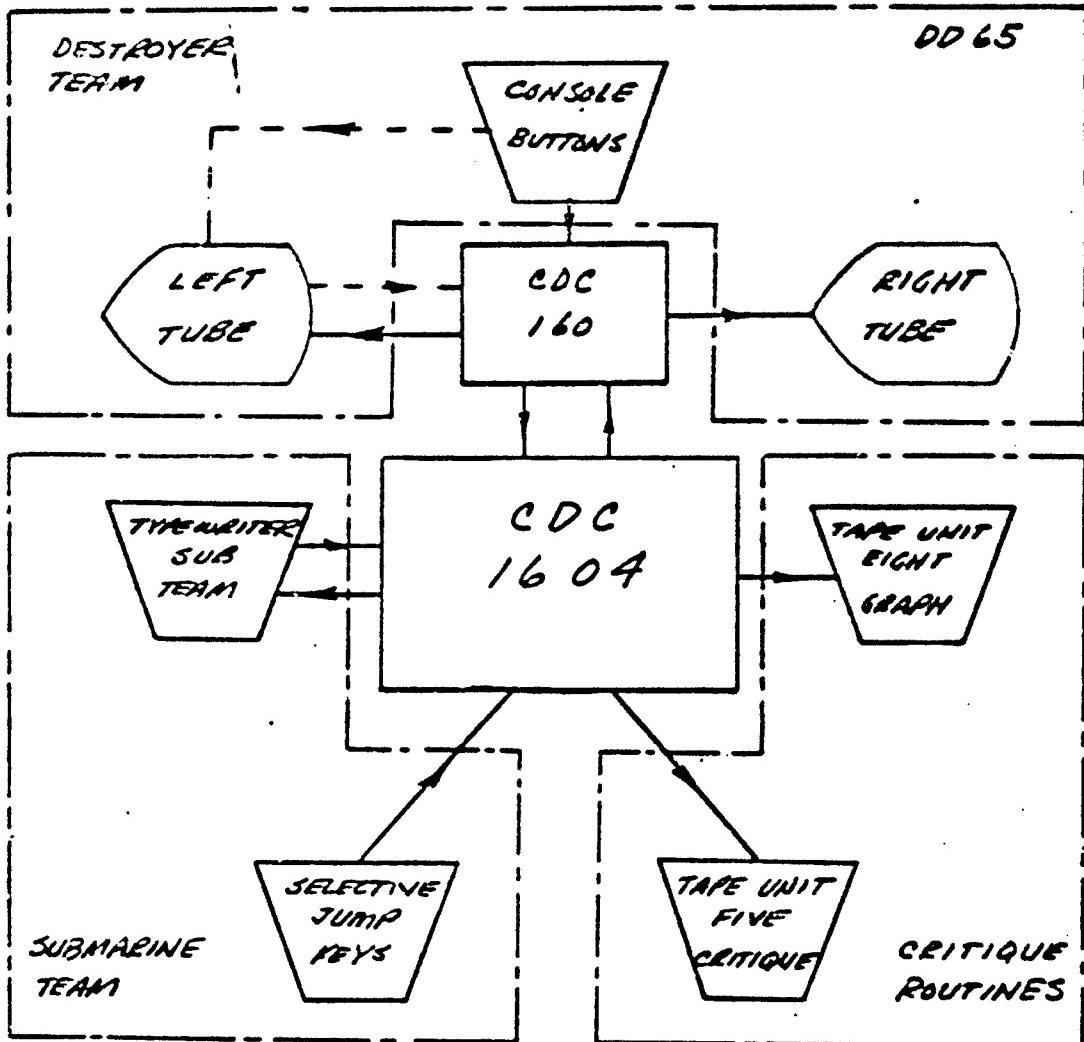


FIGURE 1

UNIFORM, NORMAL, and ERROR. The subroutine RANVAR is the basic random number generator called by these subroutines in the generation of their respective random variables. The above routines completed, only the communication routines required to link the Control Data Corporation 1604 and 160 computers remained. This requirement was satisfied by subroutines DCIRCLE, DTTRACK, PARAMS and DSTATUS (see Appendix III and Acknowledgements).

At this point the preliminary work was complete and the formulation of the initializing, iterative loop, and critique portions of the simulation was begun. These three basic steps, as discussed in section two, were incorporated into an executive control block.

#### EXECUTIVE CONTROL

The Executive Control routine is flow charted in Appendix II and consists of three major parts. The first of these parts is the initializing portion. It is made up of four blocks:

1. Inputs
2. Set Constants
3. Initialize
4. Enter Input Changes

In this subsection we shall consider the first three, leaving the latter for discussion in the subsection titled Man Machine Interface.

The Inputs block is the one in which "standard" or nominal input parameters are set. These parameters are listed below.

Number of Destroyers

Time Factor

Destroyer and Submarine Maximum Speed

Submarine Hull Parameters

Initial Positions  
Initial Courses and Speeds  
Nominal Yield of Nuclear Weapons  
Depth of Thermocline  
Maximum Range of Weapons  
Detonation Parameters for Weapons  
Random Number Generator Initializer

Any of these parameters may be changed in the enter input changes block (see Man Machine Interface subsection). These parameters were chosen as the minimal requirements necessary to produce a simulation that has some realism and yet is not too complex. The structure of this program is such that any block can be expanded to include more parameters, thereby creating a more realistic simulation.

Set Constants in a block used, as the name implies, to initialize non-changeable inputs. In this block, all the logic indicators are set to orient the game. All damage and radiation levels are set at zero. The indices for tracking, firing, and sonar contact are set at zero. This is easily followed by cross referencing Appendices I and IV.

The Initializing block begins by initializing the random number generator subroutine and then calculates the following parameters:

Water Temperature Gradient  
Submarine Crush Depth  
Operational Depth of Submarine  
Wind Direction and Velocity  
Minimum Safe Range of Weapons [6]  
Effective Sonar Range  
Sea State

These calculations are straight forward and can easily be followed by cross referencing Appendices I and IV. The block also presents input parameters of interest to the destroyer team, submarine team (if selected), and critique routine.

Subsurface nuclear bursts are divided into four classifications:

1. Very Shallow
2. Shallow
3. Deep
4. Very Deep [4]

The criteria for selection of classification are depth of burst and yield. The determination of classification of burst is made at this point in the program and an index IDEEP is set (see Appendix I). The four matrices of output data are filled with negative zero, since negative zero is programmed not to print on the display. Finally, if the role of the submarine is to be played by the computer, the basic strategy of the submarine is randomly determined (see Submarine Logic Model).

The next major part of the Executive Control Routine is the iterative loop. It is made up of eight blocks:

1. Plot Positions
2. Display Data
3. Plot Generator
4. Interactions
5. Radiation Model
6. Eater Changes
7. Submarine Logic Model
8. Time Loop

The first three of these will be considered now, while the remaining blocks will be explained in later subsections.

The Plot Positions block is utilized to transmit output information to the left tube of the DD 65 (see Figure 2). This output information consists of the following data:

- Destroyer Tracks
- Sonar Contact Plots
- Destroyer Courses, Speeds and Coordinates
- Orientation and Size of Area Displayed
- Important Messages to the Player

This information presentation is covered in more detail in the section on Man Machine Interface.

The Display Data block performs the same function with respect to the right tube of the DD 65 (see Figure 3). Figure 3 lists the data displayed by this block and for this reason it will not be listed at this point.

The final major part of the Executive Control Routine to be considered at this time is the Critique block. This block critiques the simulation by performing several tasks. The entire simulation is recorded on magnetic tape and can be reviewed at a later time. If any significant action or interaction occurs game time and the nature of the action/interaction are recorded by the Critique I block. A narrative print out is made at the conclusion of the simulation by the Critique II block. To correlate this information a graph plot of the tracks of destroyers, submarines, and all pools and clouds of radiation is made. Detail of these operations is documented in Appendix IV.

#### PLOT GENERATOR

The Plot Generator block advances all participants each time step (see Appendix II). The block then determines if there are any clouds or pools of radiation. If there are clouds, they are advanced using wind

# LEFT TUBE DISPLAY

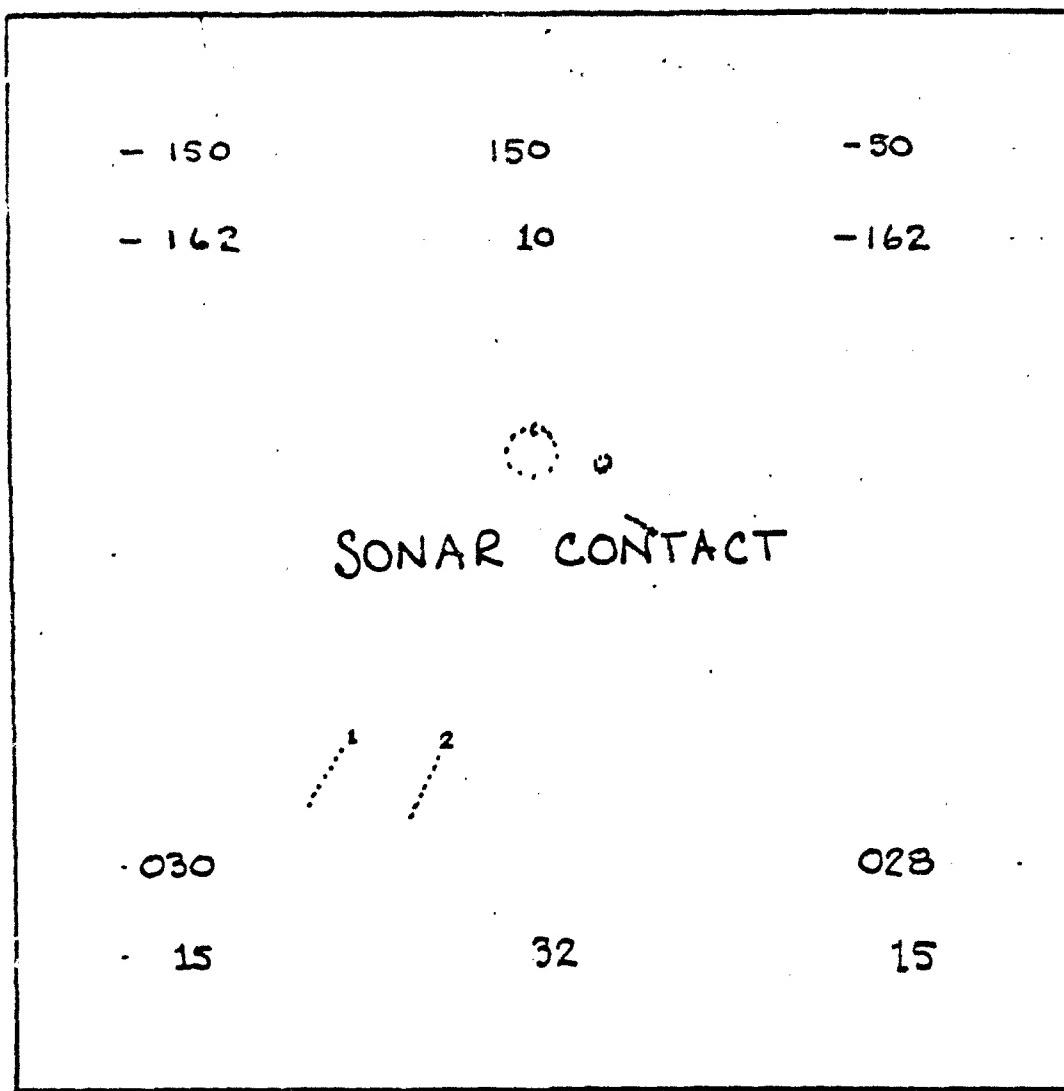


FIGURE 2

## RIGHT TUBE DISPLAY

### INPUT PARAMETERS

NUMBER OF DESTROYERS ... 2  
SIZE OF ASROC WARHEAD ... 2.0 KT  
DEPTH OF THERMOCLINE ... 120.0 FT  
MAXIMUM RANGE OF ASROC . 9000.0 YARDS  
DEPTH OF BURST ..... 700.0 FT  
SINK RATE OF WARHEAD ... 18.0 FT/SEC  
TIME FACTOR ..... 5.0  
WATER TEMP GRADIENT .... -.533 DEG/100FT

EFF SONAR RANGE	5171.	WIND DIRECTION	179
GAME TIME	42.5	WIND VELOCITY	28.
WARHEAD SIZE	2.0	SEA STATE	7
		DD1	DD2
MAX SPEED AVAIL	33.	33.	
CONTACT CLASS		1	
SONAR RANGE		4899.	
SONAR BEARING		65	
DOPPLER		1	
TARGET COURSE		2	
TARGET SPEED		22.	
FIRING SOLUTION		1	
RADIATION RATE	:	:	
RADIATION DOSE	:	:	

FIGURE 3

velocity and direction to determine motion, and the radius is computed [1]. Pools are considered to be stationary since their drift is negligible.

#### INTERACTIONS

Interaction is the largest block and for clarity has been broken into sub-blocks. These sub-blocks are:

1. Sonar Contact Model
2. Contact Tracking Model
3. Weapon Firing Model
4. Evaluation Model

The interactions block is constructed in the form of a loop that considers the interactions of each unit in succession. The organization is well documented in Appendices II and IV and will not be further explored at this time, however, the details of the four sub-blocks listed will be considered below.

#### SOFAR CONTACT MODEL

The Sonar Contact Model uses a ray path theory detection scheme in a deterministic manner [3]. This deterministic detection range then has a variance superimposed upon it. The net result is a fairly realistic sonar detection model. The major limitation of this model is that it only handles the constant temperature gradient case. The inclusion of other gradients causes the sonar detection problem to assume a much more complex nature. Included in the model is a degradation of sonar range due to excessive destroyer speed. The range and bearing given as outputs from this model have range and bearing errors included. The data, with these errors, is then utilized by the tracking and firing models. Doppler is also calculated in the model and sent to the display as:

1. Up Doppler
2. Down Doppler
3. No Doppler

#### CONTACT TRACKING MODEL

The Contact Tracking Model uses as input data the output of the Sonar Contact Model. A simple criteris, requiring three consecutive marks from the sonar model, is used to distinguish non-contacts from contacts. Once three consecutive marks are received, the model determines the contact course and speed. This is done with a simple no parameter track model that considers at least three but no more than five marks. The course and speed of the contact are determined from the first and last of these marks. This track model is unsophisticated but the error induced in the output is fairly realistic. This model is of the first that should be improved upon if more work is to be done on this simulation. The output of this model is in the form of contact course and speed. This model provides dual routing, dependent upon the track. This will be discussed further in the subsection on Block Sequencing. Determination of whether or not the target is in firing range is made just before exiting the routine.

#### WEAPON FIRING MODEL

The Weapon Firing Model takes the last position of the contact from the Sonar Model, the contact course and speed from the Tracking Model, and then determines a firing solution. The range is considered, with time of flight and sink time, and the time of burst is determined. A dead reckoning position of the target is computed from the track data and this position becomes the aim point of the weapon. The model then

calls subroutine **ERROR** from which the true fall of shot is determined. The weapon is given a reliability check in the model and if this test is failed, the player will be notified that the weapon has misfired (see Man Machine Interface subsection).

#### EVALUATION MODEL

The evaluation model initializes the pool and cloud of radiation created by the subsurface nuclear burst. To accomplish this the model takes data from the firing model for the location of ground zero and data from the inputs block for yield, depth of burst, and type of burst. The radius of the radioactive pool is then determined [4] (the cloud parameters are computed in the plot generator model).

A simple criteria for damage to the submarine is used. The lethal range is determined using submarine hull parameters, yield of warhead, and submarine depth as received from the submarine logic model. Slant range to the burst from the submarine is computed. If the submarine is within the lethal range, damage is 100%. If the submarine is outside a radius equal to twice the lethal range, damage is zero. Values of submarine damage between zero and 100% are computed by a linear relationship, then, if at any time the submarine's total damage reaches the 75% level, the game is terminated with the submarine considered as having been sunk.

#### RADIATION MODEL

The Radiation Model determines if any weapons have been detonated. If none have been detonated, the block is bypassed and the program continues. If weapons have been detonated, the model computes the distance of each destroyer from all pools and clouds of radiation. The location

and size of all pools and clouds is received from the plot generator model along with the location of the destroyers. The model then determines if the destroyers are within the perimeter of any pool or cloud. If this condition exists, the radiation level in the cloud or pool is calculated [1, 4]. The total radiation being received by each destroyer is then calculated. The radiation rate and total radiation dose for each destroyer are computed and sent to the display (see Man Machine Interface subsection).

#### SUBMARINE LOGIC MODEL

The submarine can be controlled in two ways:

1. By a submarine team.
2. By the computer.

This decision is made in the enter input changes routine. The program is such that the computer will play the role of the submarine unless the variable ISUB is set equal to one by the enter input changes routine. If ISUB is set equal to one, control of submarine depth, course, speed, and weapon firing is turned over to the submarine team. This team will receive passive sonar bearings and screw beat information from the console typewriter of the CDC 1604. They will be able to control the movements of the submarine and its weapon firing by means of the console typewriter and selective jump keys. The program will query selective jump key number two, once each cycle, to determine if orders to the submarine are to be received. If selective jump key two is set the computer will request orders (see Man Machine Interface subsection). The weapons (torpedoes) can be fired by setting selective jump key three on the CDC 1604 console (see Man Machine Interface subsection). Under this condition the simulation becomes a conflict between two teams:

1. A submarine team using the CDC 1604 console.

2. A destroyer team using the DD 65 display console.

If ISUB is unchanged by enter input changes (inputs block sets this variable equal to zero), the moves of the submarine are controlled by the computer. This being the case, two basic initial tactical situations are available to the player. The first of these places the submarine on the surface, at the origin of the playing area. The submarine knows that he has been sighted and the game proceeds. The second situation has the submarine randomly located in the upper half of the playing area. In this case the submarine's position is not known by the player. It should be noted that this is the initial situation if the submarine is to be controlled by a submarine team.

Initial situation one is selected by setting INITIAL equal to zero, while situation two is selected by setting this variable equal to one. Having selected the initial situation the computer then selects one of three basic strategies:

1. The submarine runs for it.

2. The submarine tries to transit between the two destroyers.

3. The submarine tries an end run, flanking the two destroyers.

These strategies are easily followed in Appendix II and will not be explored further at this point.

#### SEQUENCING OF PROGRAM BLOCKS

The blocks in this simulation are of two basic types:

1. A single point of exit.

2. Multiple points of exit.

The block that has a single exit point might be called a standard block, in that, the block is called upon to perform a computation but no

branching of logic is done within the block. The multiple exit block is one in which the program is routed differently depending upon logic decisions made within the block. The sonar contact block is an example of this type. In this block the routing depends upon the results of a sonar search. If contact is made, the block exists to the contact tracking block. If no contact is made, the block exits to consider the next destroyer.

The executive control routine flow chart in Appendix II illustrates the time sequencing of the major program blocks. The simulation is delayed at four points in the program which are:

1. Enter input changes routine.
2. Enter changes.
3. Submarine logic block.
4. The time loop.

The first of these interruptions takes place only during the initializing portion. At this point any change to the input parameters is made. The second of these interruptions is made once during each time step. This is the point at which destroyer team changes are sent to the CDC 1604. The third of these interruptions takes place if the situation using the submarine team has been selected. In this case the simulation may be interrupted every time step to allow the submarine team to enter changes. The fourth of these interruptions occurs each cycle and maintains the time stepping interval.

It will be noted that in the executive control routine, each block is considered in turn, no block is bypassed. In the interactions block, it will be noted that, sub-blocks are not always considered. No sonar contact by the sonar contact block causes the contact tracking block to be bypassed. The same is true of the weapon firing block if the tracking

block does not generate a satisfactory track.

#### MAN MACHINE INTERFACE

This subsection is concerned with communications, both into and out of the computer. These man-machine communications fall into four types:

1. Data to and from the destroyer team.
2. Data to and from the submarine team.
3. Data to the various modes of the critique routine.
4. Input changes.

The first type breaks into three parts:

1. Right tube information.
2. Left tube information.
3. Changes sent to the CDC 1604.

The right tube gives the destroyer team data in tabular form as illustrated in Figure 3. The left tube will display the tracks of the destroyers, any sonar contacts, and all pools and clouds of radiation. Also included in the display on the left tube is a series of windows in which data can be displayed. The windows are numbered as shown in Figure 2. Window data assignments are as listed below.

<u>Window</u>	<u>Data Assignment</u>	<u>Units</u>
1	X-coordinate of destroyer 1	100 yards
2	X-coordinate of destroyer 2	100 yards
3	Y-coordinate of destroyer 1	100 yards
4	Y-coordinate of destroyer 2	100 yards
5	Course of destroyer 1	Degrees true
6	Speed of destroyer 1	Knots

<u>Window</u>	<u>Data Assignment</u>	<u>Units</u>
7	Course of destroyer 2	Degrees true
8	Speed of destroyer 2	Knots
9	X-coordinate of left tube center	100 yards
10	Y-coordinate of left tube center	100 yards
11-14	Available for flash messages	Alfa-numeric
15	Not used	
16	Radius of display on left tube	1000 yards

Windows 5-10 and 16 are controllable from the DD 65 console by means of a discrete digital type control system. In this manner the player is able to change the destroyer course and speed, or "zoom" the display in on any point in the area of play. Let us first consider windows five through eight.

On the DD 65 console (see Figure 4) there are buttons labelled CONN DD 1 and CONN DD 2. By depressing one or both of these buttons the player is given control of the course and speed of the destroyer or destroyers selected. Near the button just selected is a group of four buttons (see Figure 4) labelled RIGHT, LEFT, FAST/UP, and SLOW/DOWN. Depressing one of these buttons will cause the appropriate variable in windows five through eight to change in the desired direction. As an example, if the player depresses both CONN buttons and then the button labelled RIGHT - both destroyers will commence a turn to the right. They will continue this turn until the button labelled RIGHT is released. The player can "come to a course" by depressing the correct button until the desired course is displayed in the respective window. The same procedure is used for changing speed. It should be noted that the windows can be changed to values that are unacceptable, such as -5 knots,

in this case the program will cause this value to be changed back into the acceptable range during the following cycle. The value of course can be increased to values greater than 360 degrees, in which case the program will correct to the acceptable value. For example, if the player increases the course to 390 degrees, the program will convert this to 030 degrees during the next cycle.

The same basic procedure is utilized to "zoom" the display to any location desired. First the player selects the button labelled SHIFT (see Figure 4). He now has control of windows nine and ten. Depressing the button labelled RIGHT will cause the display to shift to the right, LEFT accomplishes the same action but to the left, and similarly with UP and DOWN. To zoom in, the rotary switch in the upper right hand corner of the console is used (see Figure 4). This switch has six positions labelled 4, 8, 16, 32, 64, and 128. Selection of one of the six positions will cause the radius of the displayed area to be that of the value selected, in thousands of yards. As an example, selecting SHIFT and changing windows nine and ten to 240 and 150 respectively, then setting the rotary switch to eight, will cause the area centered at (24000,15000) with a radius of 8000 yards to be displayed on the left tube. Note that window 16 will show the value eight, while windows nine and ten will show 240 and 150 respectively.

Windows 11 through 14 are used to send the player the following flash messages.

SONAR CONTACT

ASROC FIRED

ASROC MISFIRE

SUB SUNK

DD SUNK

## DD65 CONSOLE ARRANGEMENT

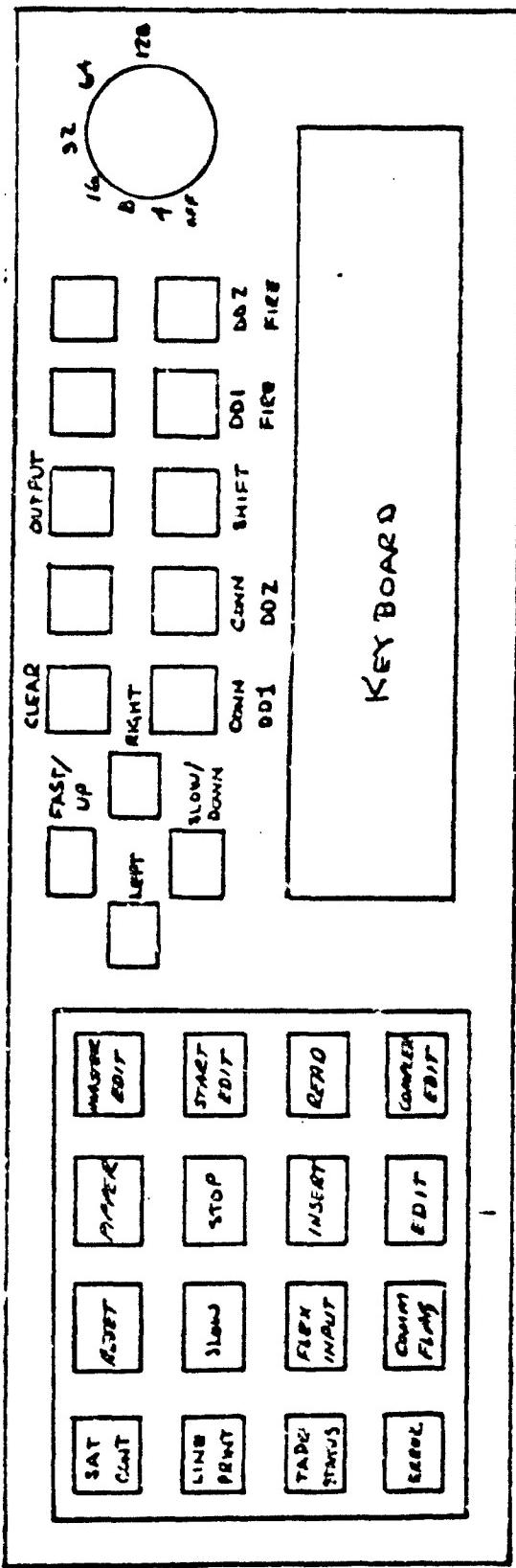


FIGURE 4

TARGET IN RANGE

TARGET TOO CLOSE TO SHOOT

TARGET OUT OF RANGE

Another method of communications is available to the destroyer team.

By depressing the DD 1 FIRE button, destroyer number one fires an ASROC.

The same procedure is used with the DD 2 FIRE button.

The next type of communications available is that of the submarine team. This is, of course, non-existent if the option is chosen in which the submarine is played by the computer. The submarine team will receive messages each cycle containing bearing and screw beat information. The submarine team may then choose to maneuver the submarine by setting selective jump key number two on the console of CDC 1604. The computer will then type COURSE ORDERS on the console typewriter. This is the indication that the computer is ready to receive course changes. If no change is desired, the old course is typed in. If a change is requested, the new course is typed in. The course typed in should be of the form 090. followed by a carriage return. This will change the course and the computer will return NEW SUB COURSE 090. This completes the course change cycle and the computer will then type SPEED ORDERS, the same procedure is used to enter speed changes. Upon completion of the speed entry the computer will return NEW SUB SPEED 15, followed by DEPTH ORDERS. The new depth is now entered, and the computer will return NEW SUB DEPTH 1050. The routine is now finished and the program continues.

The only other action available to the submarine team is that of firing torpedoes. This is accomplished by depressing selective jump key number three until the typewriter returns TORPEDO FIRED. At this time the selective jump key should be returned to the normal position, unless

DD65 CONSOLE OVERLAY

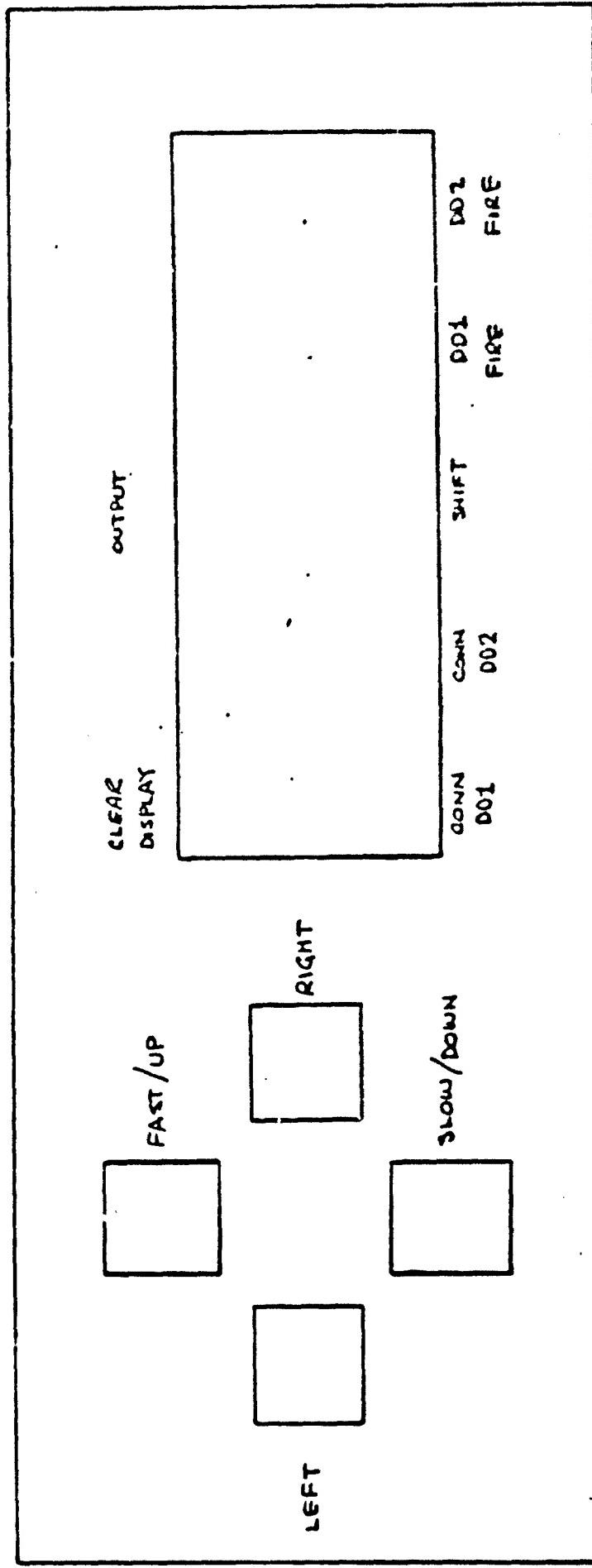


FIGURE 5

another torpedo is desired. If a hit is scored, the typewriter will return DESTROYER SUNK. When both destroyers are sunk the typewriter will return GAME OVER. At the beginning of the game, the typewriter will give the submarine team the following information:

1. Maximum speed available to submarine.
2. Maximum depth allowable.

The third type of communication is with the various critique routines.

Critique is accomplished in three ways:

1. A recording of all information on the DD 65 display is recorded by the tape unit near the DD 65.
2. A graph of the tracks of the DD's, submarine, and all radiation is made on tape unit eight of the CDC 1604 (the graph has game time recorded by each mark to aid in correlating with the various other parts of the critique routine).
3. A critique of all important events and their time is recorded on tape unit five of the CDC 1604 for print out at the conclusions of the game.

These three methods of critique, if correlated, will give an excellent "replay" of the simulation. Any communication with the program other than listed above will be accomplished as described by Leach and Perrella [5].

#### TIMING

The timing of this simulation is done by means of a time loop block. In this block, the contents of cell 5006B in the CDC 1604 is tested and stored as ICLOCK. A variable, NEXT, is generated as the sum of ICLOCK and ISTEP, the stepping interval. ISTEP is determined from another variable TFACTOR that is equal to one for real time. TFACTOR equal to three would, for example, cause the game to run at three times real time. When ICLOCK

becomes greater than or equal to ~~NEXT~~ the loop is exited and the simulation continues. This procedure is easily followed in Appendices II and IV.

## 6. CONCLUSIONS AND ACKNOWLEDGEMENTS

The considerations made in the formulation and construction of this simulation have made some observations possible. Only the basic modular structure of the simulation has been considered in detail. Each individual modular block has been designed in as simple a form as possible while maintaining some degree of realism. The simulation is sound in its general organization. New program blocks may be substituted making the simulation as realistic as desired. It is hoped that this simulation will be played with more sophisticated models and on-line equipment of greater capacity so that doctrine and tactics in the area of self-inflicted nuclear environment may be explored.

It was found that the general purpose computer language was completely satisfactory for the construction of this simulation. The modularization technique made the logical organization of the simulation straight forward and is recommended for use in future simulations.

The author wishes to express his appreciation to Professor Mitchell L. Cotton and Professor Alvin F. Andrus for their aid and encouragement in the preparation of this thesis. In addition, the author would like to thank Miss Patricia Hoang for her assistance in programming the linking subroutines and LCDR Richard E. DeWinter for his comments in proofreading the manuscript.

## BIBLIOGRAPHY

1. Huebsch, I. O. A Model for Computing Base-surge Dose-rate Histories for Underwater Nuclear Bursts (U). USNADL-TR-653, 1963. (CCWF)
2. Johnson, J. E. Methods for Digital Simulation of Military Conflict Situations. USNPGS Thesis, 1965.
3. Kinsler, L. E. and Frey, A. R. Fundamentals of Acoustics. John Wiley and Sons, New York, 1962.
4. Ksanda, C. P. Analysis and Prediction of the Properties of Diffusing Radioactive Pools from Nuclear Explosions in the Ocean (U). USNADL-TR-725, 1963. (CCWF)
5. Leach, G.H. and Perralla, A. J. A Satellite Computer System for On-Line Analysis, Control and Display. USNPGS Thesis, 1964.
6. Sulit, R. A. Analytical Model and Proposed Umpiring Procedures for Ship Damage and Combat Ineffectiveness from Initial Nuclear Weapons Effects (U). USNADL-TR-720, 1964. (CCWF)
7. Fundamentals of War Gaming, United States Naval War College, 2nd ed., November 1961.

## APPENDIX I

### LIST OF VARIABLES

This Appendix contains a listing of the variables used in the simulation MULNUC1, arranged in alphabetical order.

<u>Variable</u>	<u>Definition</u>
AROCMAX	Maximum range of the ASROC, in yards. This is an input parameter, set equal to 9,000 yards by the inputs block and can be changed in the change inputs block.
AROCMIN	Minimum safe range for the ASROC, in yards. This value is computed in the initializing block, and is a function of warhead size.
B	Last bearing the submarine held of the nearest destroyer, in degrees true.
BNPTS	A dummy variable used in the contact tracking block for the determination of contact speed.
CENTERB	The course such that the submarine will split the channel between the destroyers, in degrees true.
CLOUD <i>i</i> ( <i>r</i> )	Radius of the <i>i</i> <sup>th</sup> cloud of radiation, in yards.
CLOUDX( <i>I</i> )	X-coordinate of the <i>i</i> <sup>th</sup> cloud of radiation with respect to the main coordinate system, in yards.
CLOUDY( <i>I</i> )	Y-coordinate of the <i>i</i> <sup>th</sup> cloud of radiation with respect to the main coordinate system, in yards.
CONTB( <i>I</i> )	Bearing of sonar contact of <i>i</i> <sup>th</sup> destroyer, in degrees true.
CONTC( <i>I</i> )	Contact course, as computed by the contact tracking block, in degrees true.
CONTR( <i>I</i> )	Sonar range to contact, as measured by the <i>i</i> <sup>th</sup> destroyer.
CONTS( <i>I</i> )	Contact speed as computed by the contact tracking model, in knots.
CONTX( <i>I</i> )	X-coordinate of the <i>i</i> <sup>th</sup> destroyer's contact, in yards, with respect to the main coordinate system.
CONTY( <i>I</i> )	Y-coordinate of the <i>i</i> <sup>th</sup> destroyer's contact, in yards, with respect to the main coordinate system.

<u>Variable</u>	<u>Definition</u>
D	Difference between the true bearing to the submarine from the destroyer and the submarine's true course, in degrees. This is used in the determination of doppler.
DA	Absolute value of D.
DAMAGE	Percent damage to the submarine from the current detonation.
DAMAGET	Cumulative damage to the submarine, in percent.
DDC(I)	Course of the $i^{\text{th}}$ destroyer, in degrees true.
DDS(I)	Speed of the $i^{\text{th}}$ destroyer, in knots.
DDS MAX(I)	Maximum speed available to the $i^{\text{th}}$ destroyer, in knots.
DDX(I)	X-coordinate of the $i^{\text{th}}$ destroyer, in yards, with respect to the main coordinate system.
DDY(I)	Y-coordinate of the $i^{\text{th}}$ destroyer, in yards, with respect to the main coordinate system.
DETR	Detection range, in yards. This is a random variable with mean DETRM and normally distributed with sigma of .3 times DETRM.
DETRM	Mean detection range, in yards. A function of GRAD SUBD.
DISTC(I,J)	Distance of the $i^{\text{th}}$ destroyer from the center of the $j^{\text{th}}$ cloud of radiation.
DISTP(I,J)	Distance of the $i^{\text{th}}$ destroyer from the center of the $j^{\text{th}}$ pool of radiation.
DOB	Depth of burst, in feet, of ASROC warhead.
DR	Advance of the destroyer considered, in yards, between marks as considered by the tracking model.
DTIME	Total time delay from the firing of an ASROC and the detonation, in seconds. This includes time of flight and time of sinking.
DUMMY	A dummy variable used throughout the program.
DX	East-west advance of the destroyer considered, in yards, between marks as considered by the tracking model.

<u>Variable</u>	<u>Definition</u>
DY	North-south advance of the destroyer considered, in yards, between marks as considered by the tracking model.
ESR	Effective sonar range, in yards. Computed by the initializing block.
GAD	Water temperature gradient, in degrees per hundred feet of depth.
GTIME	Game time, in minutes. At the start of the game GTIME is zero.
GZX	X-coordinate of ground zero for the detonation considered, in yards, with respect to the main coordinate system.
GZY	Y-coordinate of ground zero for the detonation considered, in yards, with respect to the main coordinate system.
HULL	Submarine hull thickness, in inches of steel.
IASROC	Hull number of the destroyer that fired the last ASROC. If no ASROC has been fired, IASROC is zero.
ICLASS(I)	Sonar classification of the $i^{\text{th}}$ destroyer's contact. 0 - no contact, 1 - possible submarine, 2 - probable submarine.
ICLOCK	Contents of location 5006B in the CDC 1604. The CDC 1604 steps this cell once every second.
ICIMCX(I)	A dummy vector of X-coordinates of points to transmit circles to the display.
ICIRCY(I)	A dummy vector of Y-coordinates of points to transmit circles to the display.
ICONTB(I)	Fixed point version of CONTB(I), used to transmit to the display.
ICONTC(I)	Fixed point version of CONTC(I), used to transmit to the display.
ICRIT#	A series of critique indicators. 0 - pass.
1	1 - sonar contact
2	1 - ASROC fired

<u>Variable</u>	<u>Definition</u>
ICRIT#	
3	1 - torpedo fired
4	1 - ASROC misfire
5	1 - target in range
6	1 - sub sunk
7	1 - destroyer sunk
8	1 - target too close to shoot at
9	1 - target out of ASROC range
11	1 - game is a draw, submarine escaped
12	1 - submarine wins, destroyer 1 sunk with submarine escaping
13	1 - submarine wins, destroyer 2 sunk with submarine escaping
14	1 - submarine wins, both destroyers sunk
15	1 - destroyers win, submarine sunk by destroyer 1
16	1 - destroyers win, submarine sunk by destroyer 2
17	1 - submarine wins by transiting between the destroyers
IDDC(I)	Fixed point version of DDC(I), used to transmit to the display.
IDDS(I)	Fixed point version of DDS(I), used to transmit to the display.
IDDX(I)	Fixed point version of DDX(I), used to transmit to the display.
IDDY(I)	Fixed point version of DDY(I), used to transmit to the display.
IDEEP	An index used to indicate the classification of nuclear burst. 1 - very shallow, 2 - shallow, 3 - deep, 4 - very deep.
IDOPLER(I)	An index used to indicate the doppler of the $i^{\text{th}}$ destroyer's contact. 0 - no doppler, 1 - up doppler, 2 - down doppler.

<u>Variable</u>	<u>Definition</u>
IEND	An index used to indicate game over. 0 - game not over, 1 - game over.
IFIRE	An index used to indicate if the submarine has fired a torpedo. 0 - torpedo not active, 1 - torpedo still active.
ILOGIC	A logic index used in the submarine model.
IONE	The lowest destroyer hull number. Equal to 1 at the beginning of the game. If destroyer number one is sunk the IONE is equal to two.
INITIAL	An index of the initial situation. 0 - submarine at the origin, 1 - submarine randomly distributed in the upper half of the playing area.
IR	Radius of display area, in thousands of yards.
IRANDOM	A random number selected to initialize the random number generator. This number must be an odd integer in the interval 1 to 67108863.
ISBOOT1	A flag used to activate the firing sequence for destroyer 1. 0 - do not shoot, 1 - shoot.
ISBOOT2	A flag used to activate the firing sequence for destroyer 2. 0 - do not shoot, 2 - shoot.
ISOL(I)	A progressive index of the quality of the firing solution the $i^{\text{th}}$ destroyer has on its target. 0 - no solution, thru 5 - best solution.
ISS	Sea state.
ISTEP	Fixed point version of TSTEP.
ISTRAT	Submarine basic strategy when the submarine is controlled by the computer. 0 - run for it, 1 - go up the middle, 2 - end run.
ISUBC	Fixed point version of SUBC.
ITURN	Logical index that records the submarine's initial turn. 0 - left turn, 1 - right turn.
IX0	X-coordinate of center of displayed area, in hundreds of yards with respect to the main coordinate system.
IXDD(I,J)	A dynamic table of track data, recording the X-coordinate of the $j^{\text{th}}$ mark of the $i^{\text{th}}$ destroyer. The maximum value of $j$ is eight. Entries are in yards, with respect to the main coordinate system.

<u>variable</u>	<u>Definition</u>
IIXDD1(I)	This variable is equal to IXDD(1, I).
IIXDD2(I)	This variable is equal to IXDD(2, I).
IXSUB(I,J)	A dynamic table of track data, recording the X-coordinate of the jth mark of the ith destroyer's sonar contact. The maximum value of j is eight. Entries are in yards, with respect to the main coordinate system.
IXSUB1(I)	This variable is equal to IXSUB(1,I).
IXSUB2(I)	This variable is equal to IXSUB(2,I).
IY0	Y-coordinate of center of displayed area, in hundreds of yards with respect to the main coordinate system.
IYDD(I,J)	A dynamic table of track data, recording the Y-coordinate of the jth mark of the ith destroyer. The maximum value of j is eight. Entries are in yards, with respect to the main coordinate system.
IYDD1(I)	This variable is equal to IYDD(1,I).
IYDD2(I)	This variable is equal to IYDD(2,I).
IYSUB(I,J)	A dynamic table of track data, recording the Y-coordinates of the jth mark of the i <sup>th</sup> destroyer's sonar contact. The maximum value of j is eight. Entries are in yards, with respect to the main coordinate system.
IYSUB1(I)	This variable is equal to IYSUB(1, I).
IYSUB2(I)	This variable is equal to IYSUB(2, I).
IWINDD	Fixed point version of WINDD, used to transmit data to the display.
MARKS(I)	Number of continuous marks, up to five, the i <sup>th</sup> destroyer has on his sonar contact.
N	A dummy variable.
NCOUNTER	Number of constant bearing the submarine has on the nearest destroyer.
NEXT	Time of the next cycle, in seconds of computer time.
NMARKS(I)	Same as MARKS(I) except NMARKS(I) does not stop at five.

<u>Variable</u>	<u>Definition</u>
NOSHOT	Index to limit destroyers to one active weapon at a time. 0 - there are no active weapons, alright to shoot, 1 - there is an active weapon, cannot shoot.
NPTS	Index used to control display data until there are eight point available on destroyer tracks.
NPTS1	A dummy variable.
NPTS2	A dummy variable.
NRDD	The highest destroyer hull number. Equal to 2 at the beginning of the game. If destroyer number two is sunk then NRDD is equal to one.
NSHOTS	Total number of ASROC's fired during the game.
PWIT	A random variable used to determine if a torpedo, fired by the submarine, hit the destroyer.
POOLR(I)	Radius of the $i^{\text{th}}$ pool of radiation, in yards.
POOLX(I)	X-coordinate of the $i^{\text{th}}$ pool of radiation, in yards, with respect to the main coordinate system.
POOLY(I)	Y-coordinate of the $i^{\text{th}}$ pool of radiation, in yards, with respect to the main coordinate system.
R	Radius of the displayed area, in yards.
REL	A random variable used to determine ASROC reliability.
RADDOSE(I)	Total radiation dose the $i^{\text{th}}$ destroyer has been exposed to during the game, in roentgens.
RADRATE(I)	The rate at which the $i^{\text{th}}$ destroyer is receiving radiation, in roentgens/hr.
RANDOM	The random variable used to link the various random generators.
RANGE	A dummy variable used as a temporary storage in range calculations.
LETHAL	The lethal range of the ASROC warhead, in yards.
SAFETY	Safety factor used to compute submarine maximum operating depth as a function of the crush depth of the hull.
SB(I)	Screw beat count of the $i^{\text{th}}$ destroyer as measured by the submarine.

<u>Variable</u>	<u>Definition</u>
SIGMA	A dummy variable used in the calling of normally distributed random variables.
SR	Sink rate of the ASROC warhead, in feet per second.
SSB(I)	The actual bearing of the submarine from the $i^{\text{th}}$ destroyer, in degrees true.
STRESS	Yield strength of the steel used in the submarine hull, measured in thousands of pounds per square inch.
SUBC	The true submarine course, in degrees true.
SUBD	The actual submarine depth in feet.
SUBDMAX	The maximum allowable operational depth of the submarine, in feet.
SUBS	The actual speed of the submarine, in knots.
SUBSMAX	The maximum speed available to the submarine, in knots.
SUEX	The actual submarine X-coordinate with respect to the main coordinate system, in yards.
SUBY	The actual submarine Y-coordinate with respect to the main coordinate system, in yards.
SUM	A dummy variable used in the computation of radiation dose.
TBURST(I)	Time of detonation of the $i^{\text{th}}$ ASROC warhead, measured in game time.
TEMP	A dummy variable.
TEMP1	A dummy variable.
TEMP2	A dummy variable.
TEMP3	A dummy variable.
TEMP4	A dummy variable.
TEMP5	A dummy variable.
TEMPCR(I,J)	The radiation rate received by the $i^{\text{th}}$ destroyer from the $j^{\text{th}}$ cloud of radiation, in roentgens per hour.
TEMPPR(I,J)	The radiation rate received by the $i^{\text{th}}$ destroyer from the $j^{\text{th}}$ cloud of radiation, in roentgens per hour.

<u>Variable</u>	<u>Definition</u>
TFACOR	A time factor, 1 - real time, 2 - double time, .5 - half time.
THERMO	Depth of the thermocline, in feet.
THETA	A dummy variable used in the computation of courses and bearings.
TINTER	Time to intercept of a torpedo fired by the submarine, in minutes.
TLOGIC	A time storage point used to control timed logic.
TOB	Time of burst, this includes time of flight and sink time of the ASROC warhead, measured in minutes of game time.
TOF	Time of flight of the ASROC warhead, in seconds.
TOS	Sink time of the ASROC warhead, in seconds.
TSTEP	The time step for each cycle of the game, in seconds.
TXDD(I,J)	A temporary storage of the $j^{\text{th}}$ past position X-coordinate of the $i^{\text{th}}$ destroyer used in preparing track data for the display, measured in yards.
TYDD(I,J)	A temporary storage of the $j^{\text{th}}$ past position Y-coordinate of the $i^{\text{th}}$ destroyer used in preparing track data for the display, measured in yards.
TXSUB(I,J)	A temporary storage of the $j^{\text{th}}$ past position X-coordinate of the $i^{\text{th}}$ destroyer's sonar contact used in preparing track data for the display, measured in yards.
TYSUB(I,J)	A temporary storage of the $j^{\text{th}}$ past position Y-coordinate of the $i^{\text{th}}$ destroyer's sonar contact used in preparing track data for the display, measured in yards.
VEL	The relative velocity of the torpedo fired by the submarine, with respect to the destroyer target, in knots.
W1	A dummy variable used to determine the destroyer that is closest to the submarine.
W2	A dummy variable used to determine the destroyer that is closest to the submarine.
WINDD	Wind direction in degrees true.
WINDV	Wind velocity, in knots.

<u>Variable</u>	<u>Definition</u>
X(I,J)	The X-coordinate of the $i^{\text{th}}$ destroyer's $j^{\text{th}}$ continuous mark on its sonar contact, in yards, with respect to the main coordinate system.
XDD(I,J)	The X-coordinate of the $i^{\text{th}}$ destroyer $j-1$ steps ago, in yards, with respect to the main coordinate system.
X0	The X-coordinate of the center of the displayed area, in yards, with respect to the main coordinate system.
XTEMP	A dummy variable used for temporary storage of X-coordinates.
Y(I,J)	The Y-coordinate of the $i^{\text{th}}$ destroyer's $j^{\text{th}}$ continuous mark on its sonar contact, in yards, with respect to the main coordinate system.
YDD(I,J)	The Y-coordinate of the $i^{\text{th}}$ destroyer $j-1$ steps ago, in yards, with respect to the main coordinate system.
YIELD	The yield of the ASROC warhead in kilotons.
Y0	The Y-coordinate of the center of the displayed area, in yards, with respect to the main coordinate system.
ZZX	A vector of destroyer and submarine X-coordinates used by the graph plot routine in Critique I.
ZZY	A vector of destroyer and submarine Y-coordinates used by the graph plot routine in Critique I.

**APPENDIX II**  
**LOGIC FLOW DIAGRAMS**

This Appendix contains a series of logic flow diagrams as listed below.

Executive Control  
Plot Generator  
Interactions  
Radiation Model  
Submarine Logic Model  
Run for It  
Up the Middle  
End Run  
Submarine Team Control  
Sonar Contact Model  
Contact Tracking Model  
Weapon Firing Model  
Evaluation Model  
Critique I  
Critique II

It will be noted that the statement number from the program listing is shown in the upper right hand corner of each symbol in the block diagram. This should aid in correlating this Appendix, with Appendix IV.

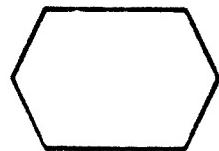
## FLOW CHART SYMBOLS



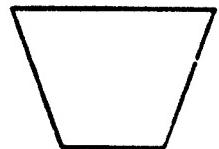
A connector or terminal.



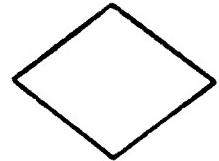
An offpage connector.



A predefined process or module/subroutine.  
A more detailed flow chart of this subroutine is also included.



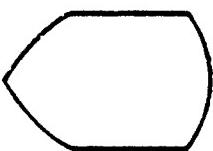
Input/output other than display.



Decision.

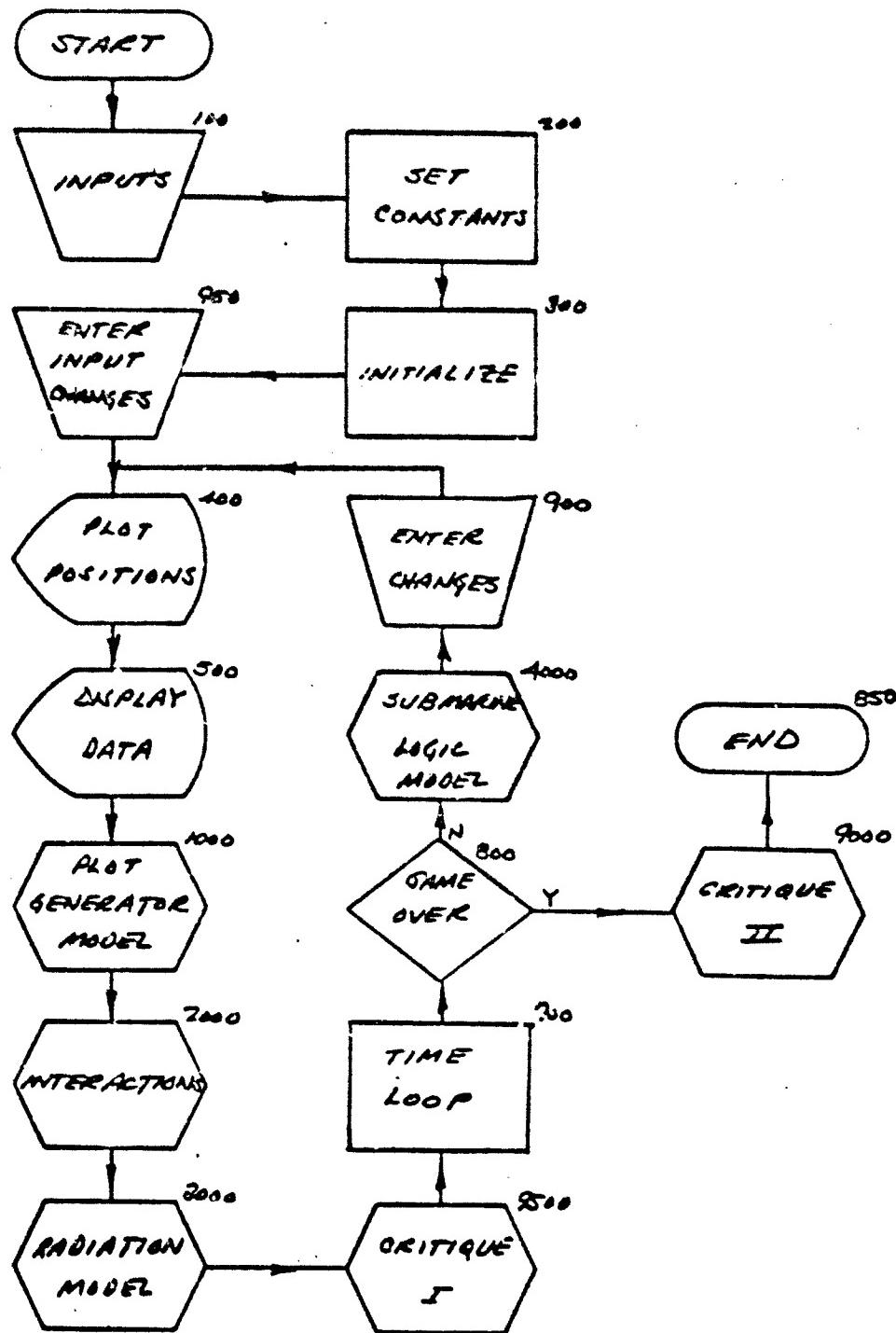


Processing, annotation.

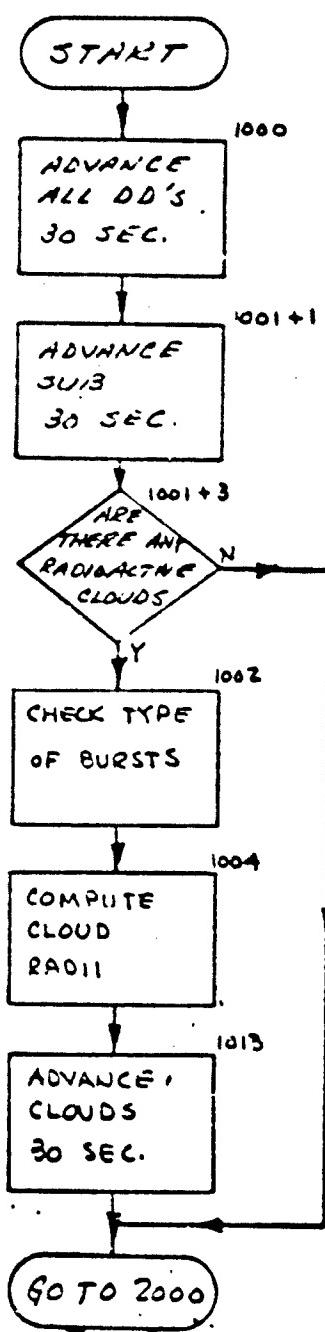


Display.

# EXECUTIVE CONTROL

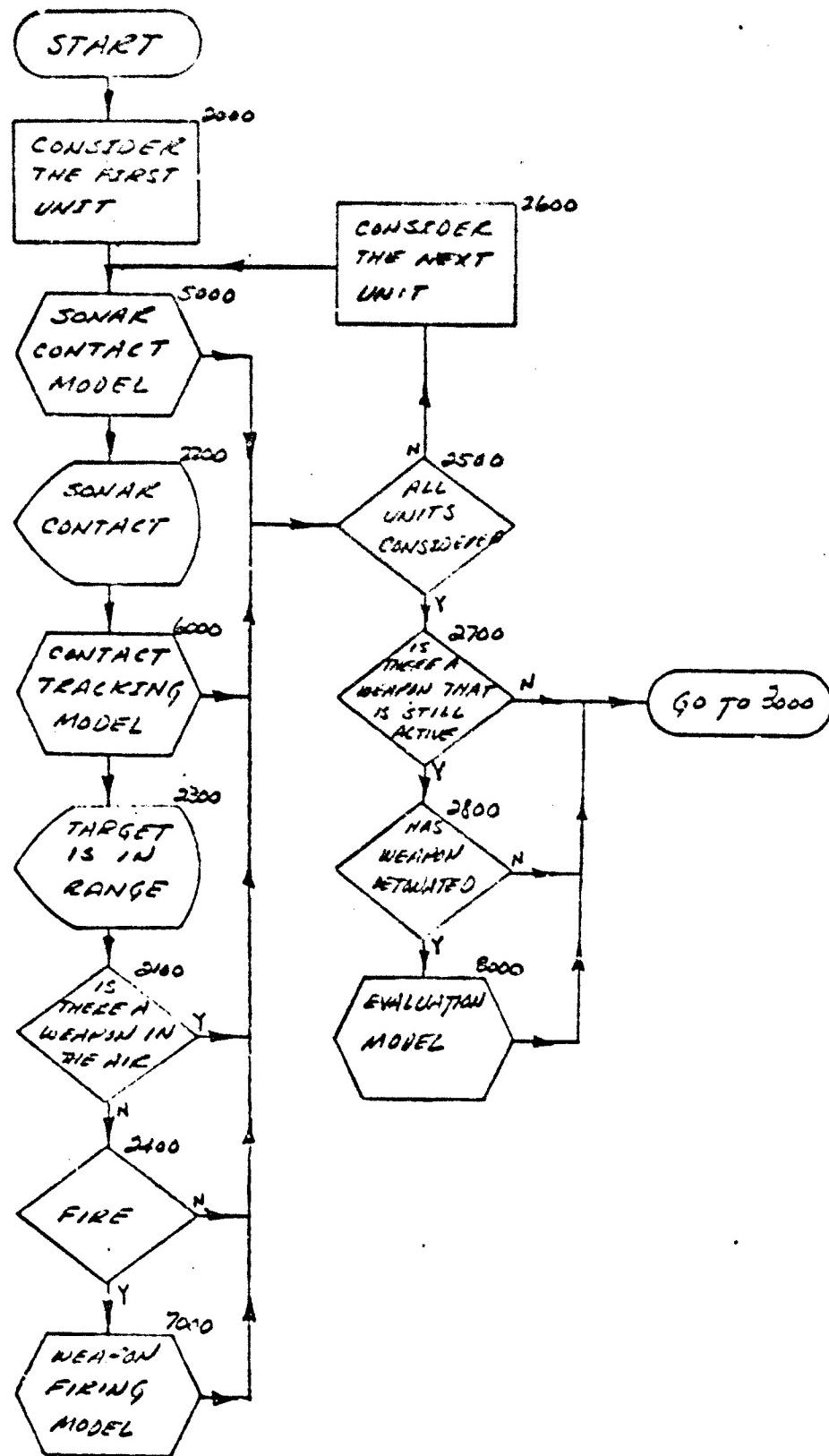


# PLOT GENERATOR MODEL

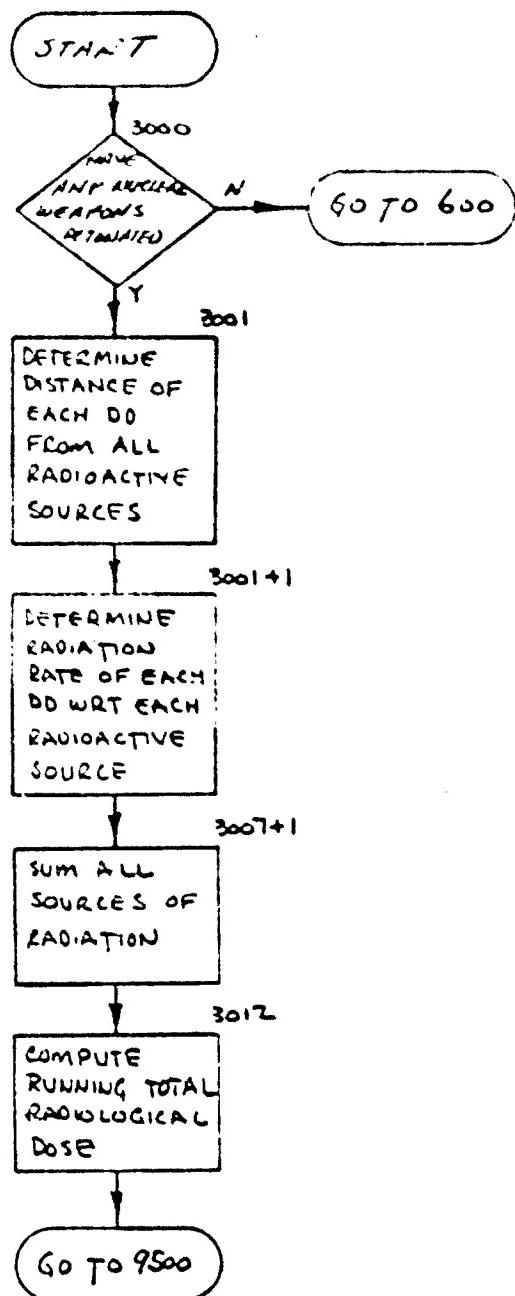


53

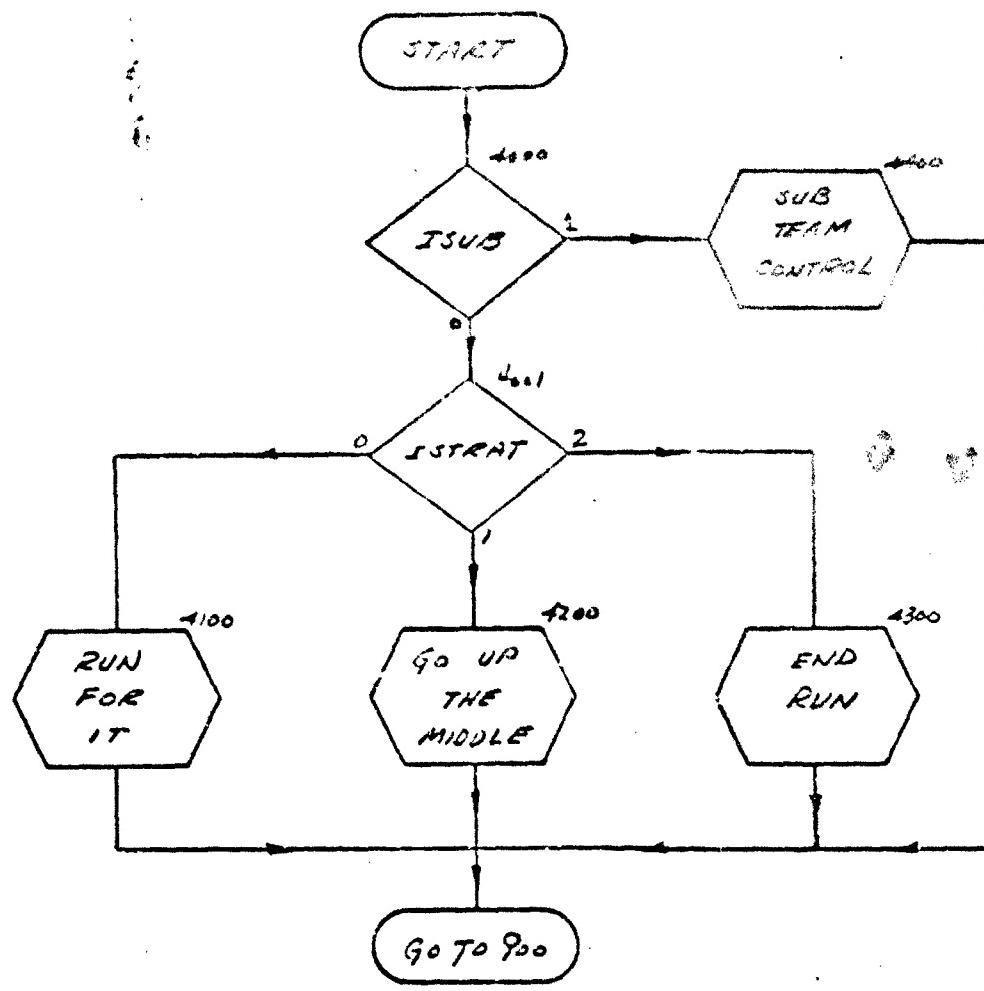
# INTERACTIONS



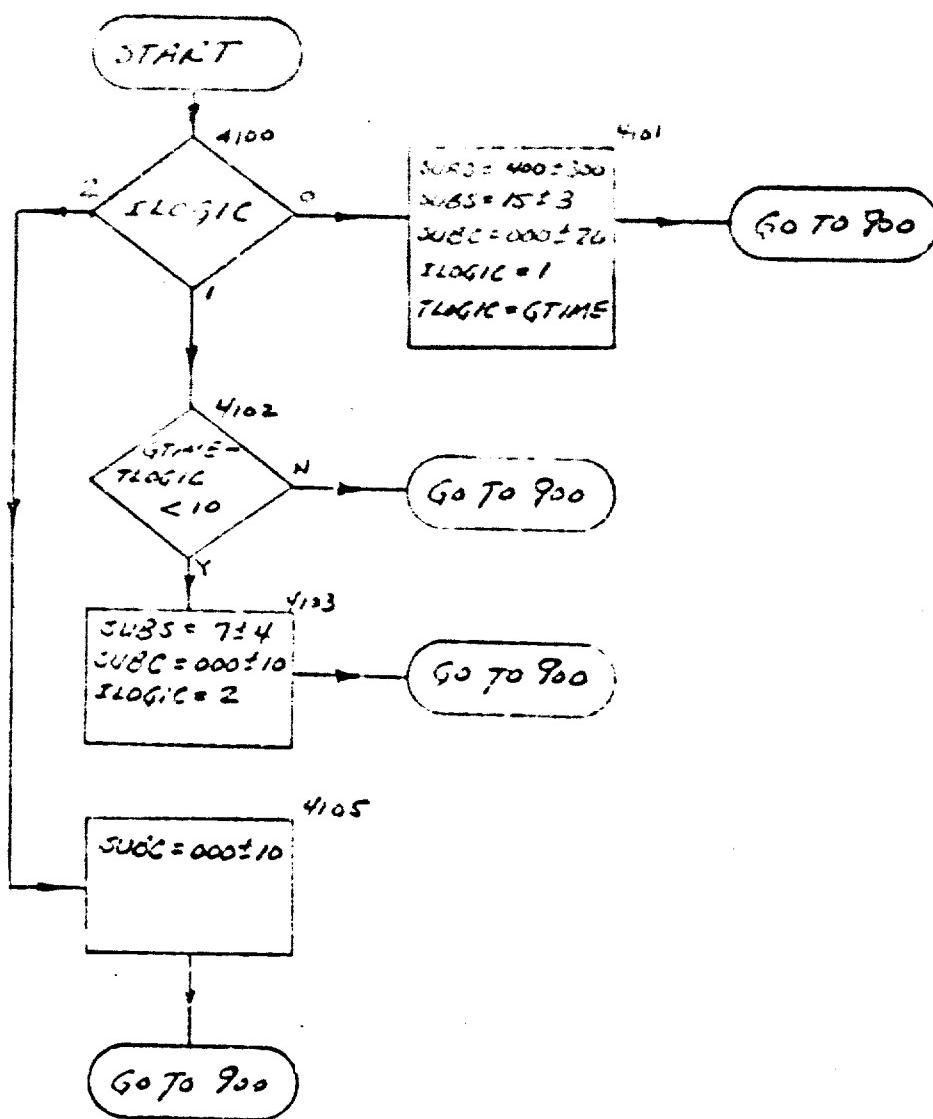
# RADIATION MODEL



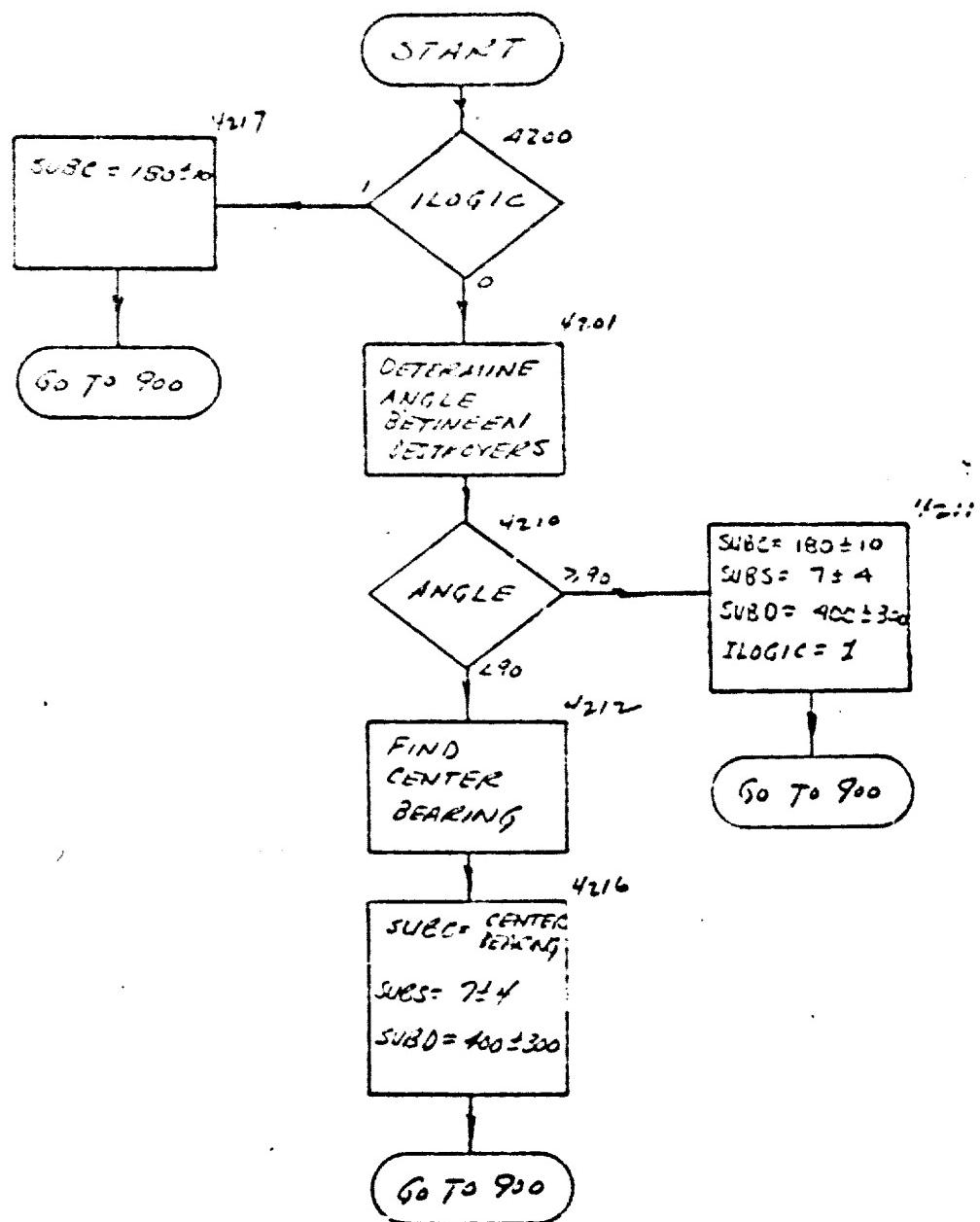
# SUBMARINE LOGIC MODEL



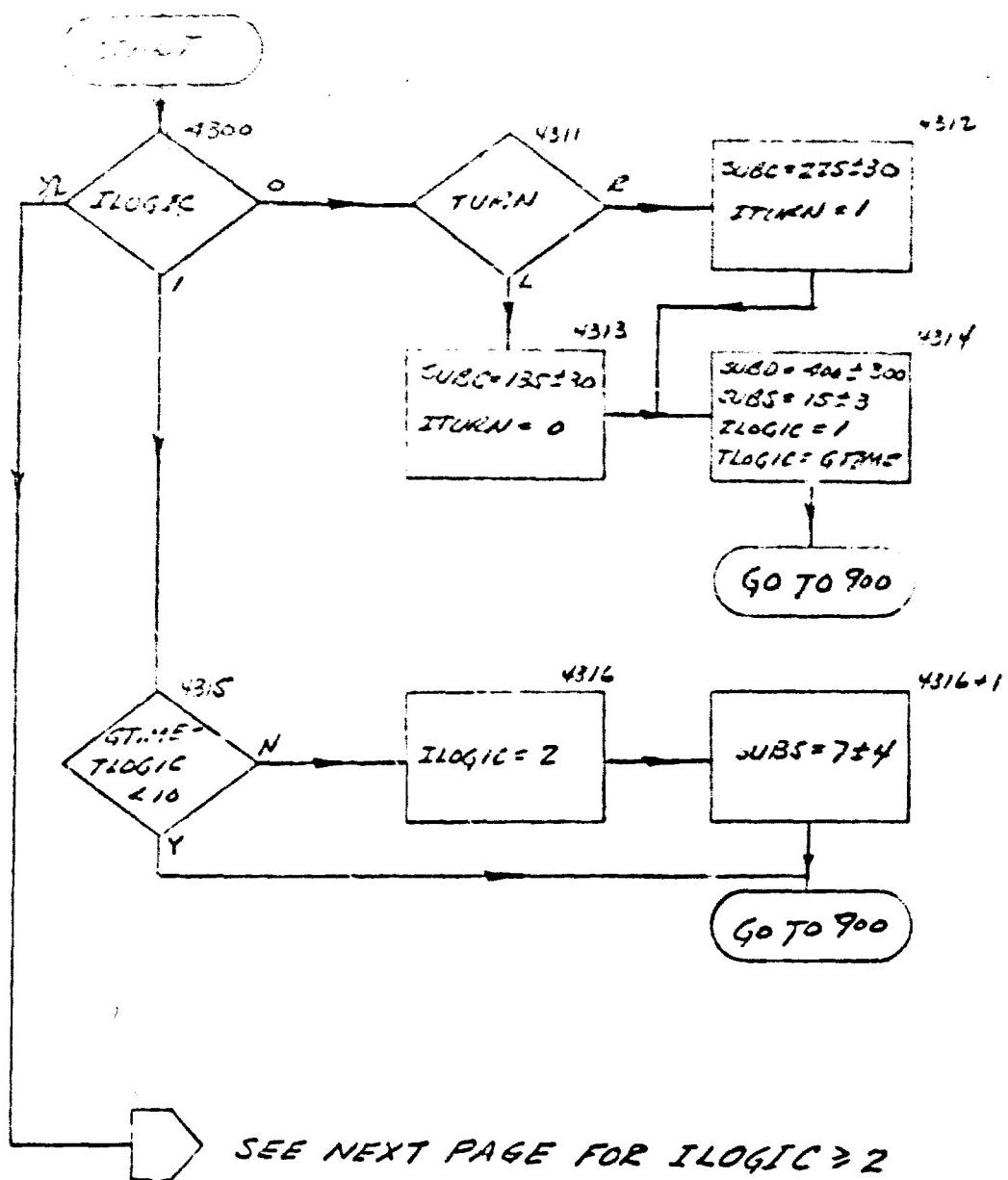
# RUN FOR IT



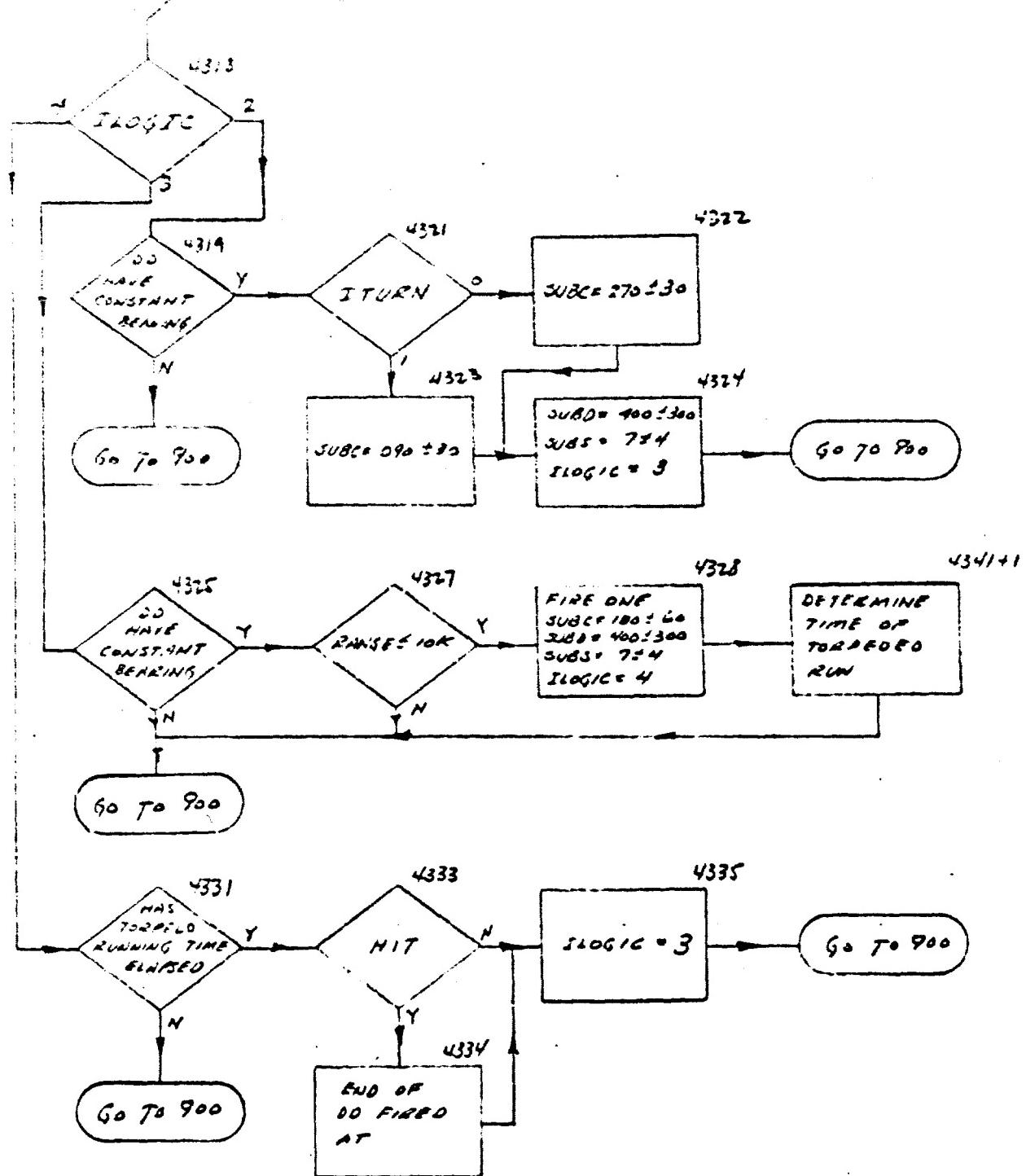
# UP THE MIDDLE



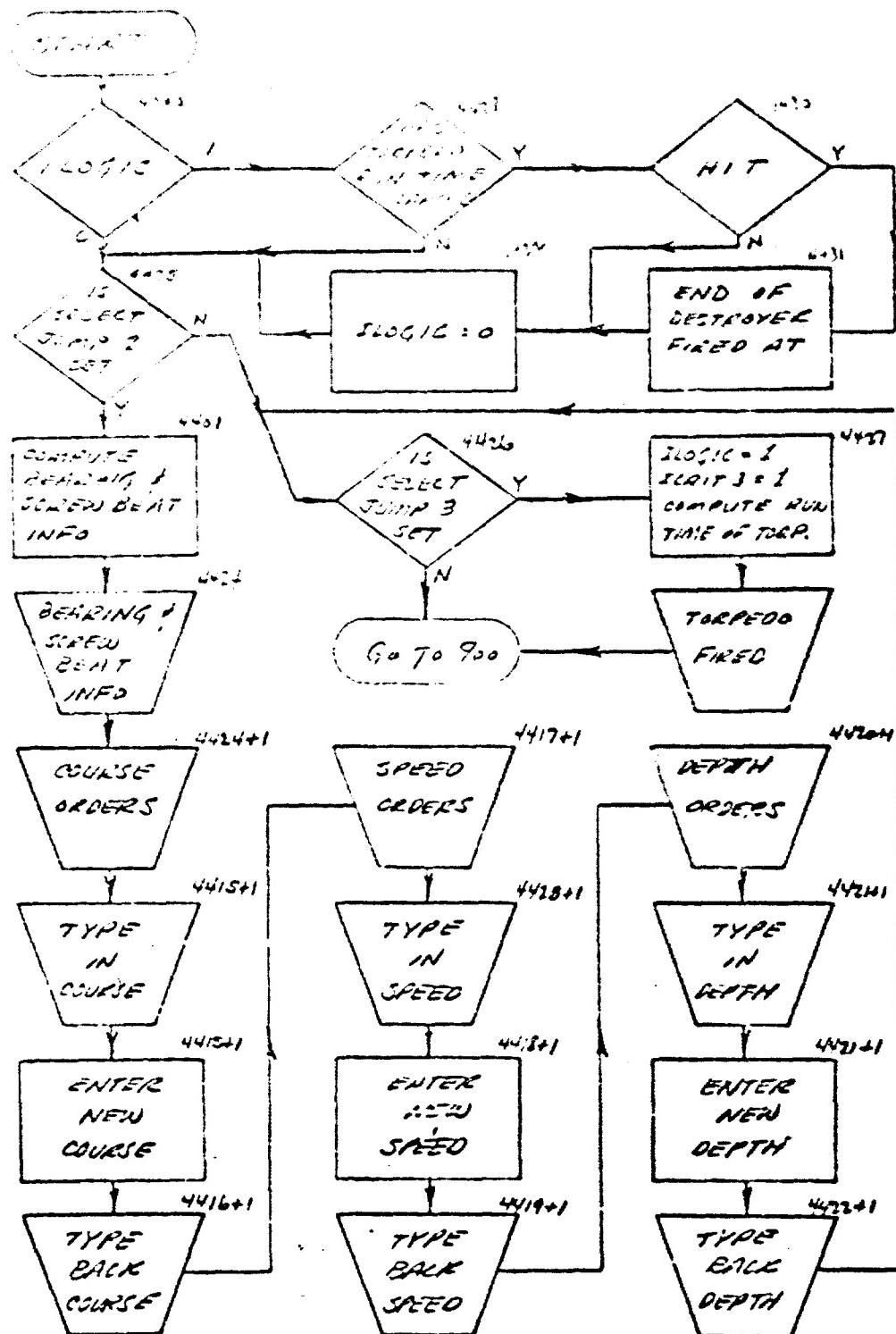
Reproduced From  
Best Available Copy



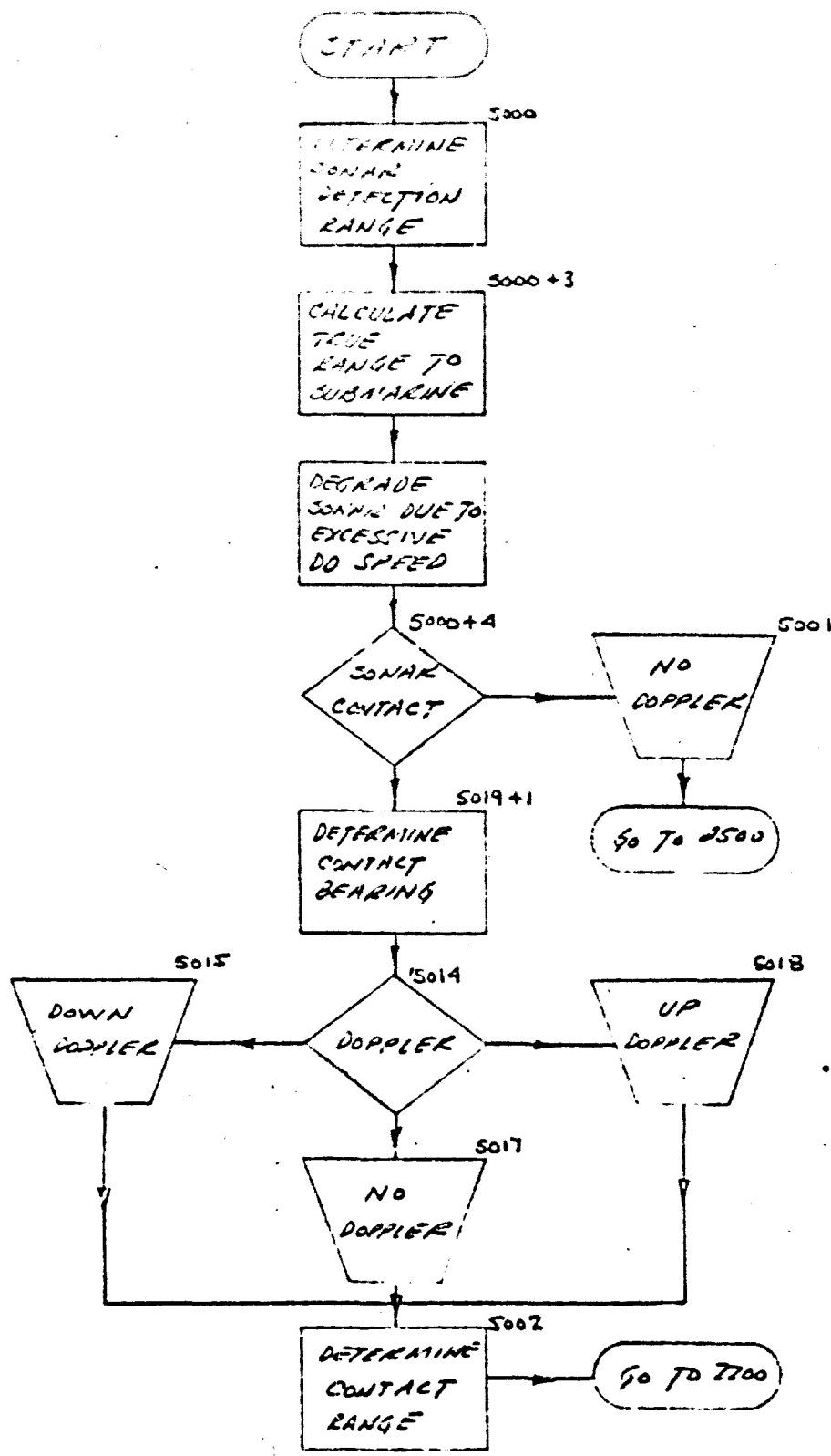
CONTINUED FROM PRECEDING PAGE



# SUBMARINE TELM CONTROL

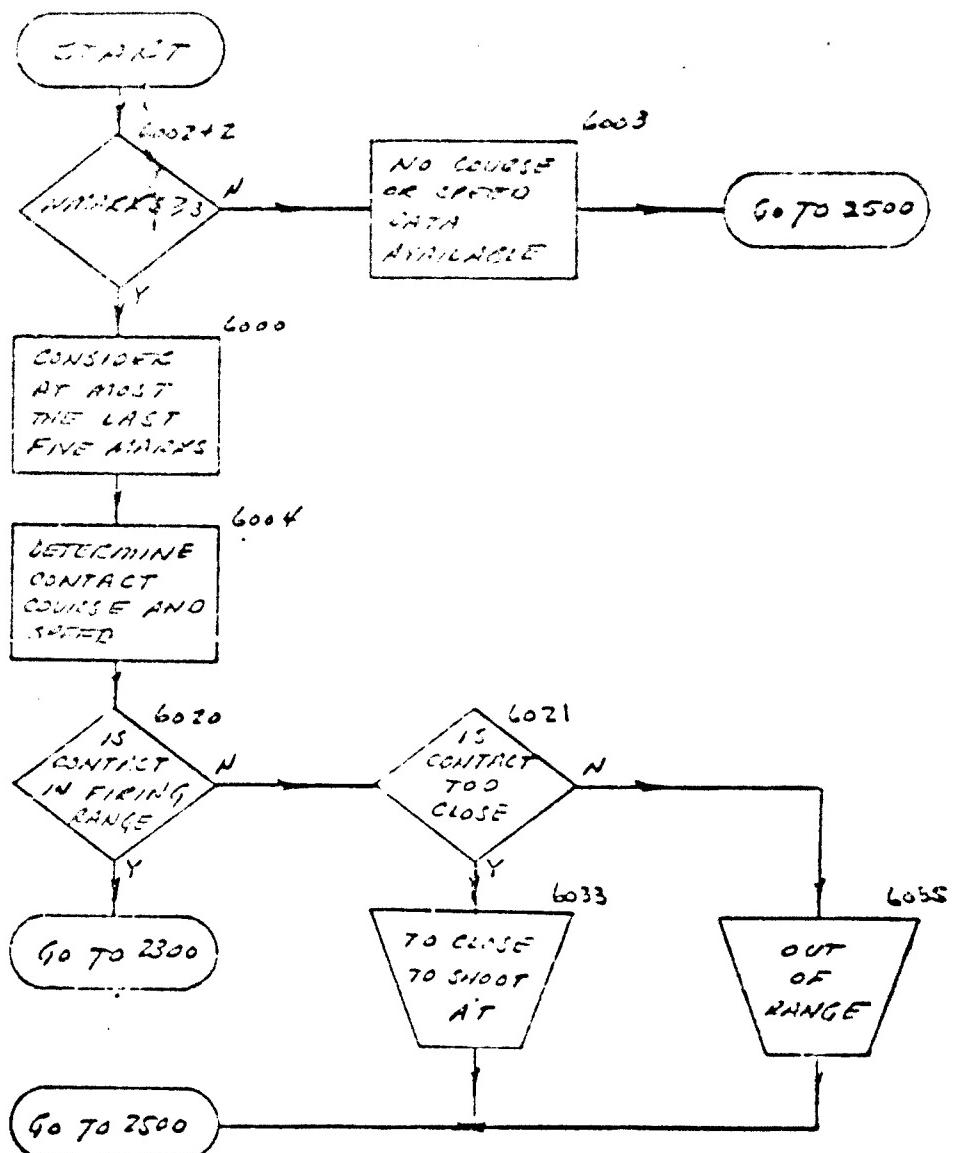


# SONAR CONTACT MODEL

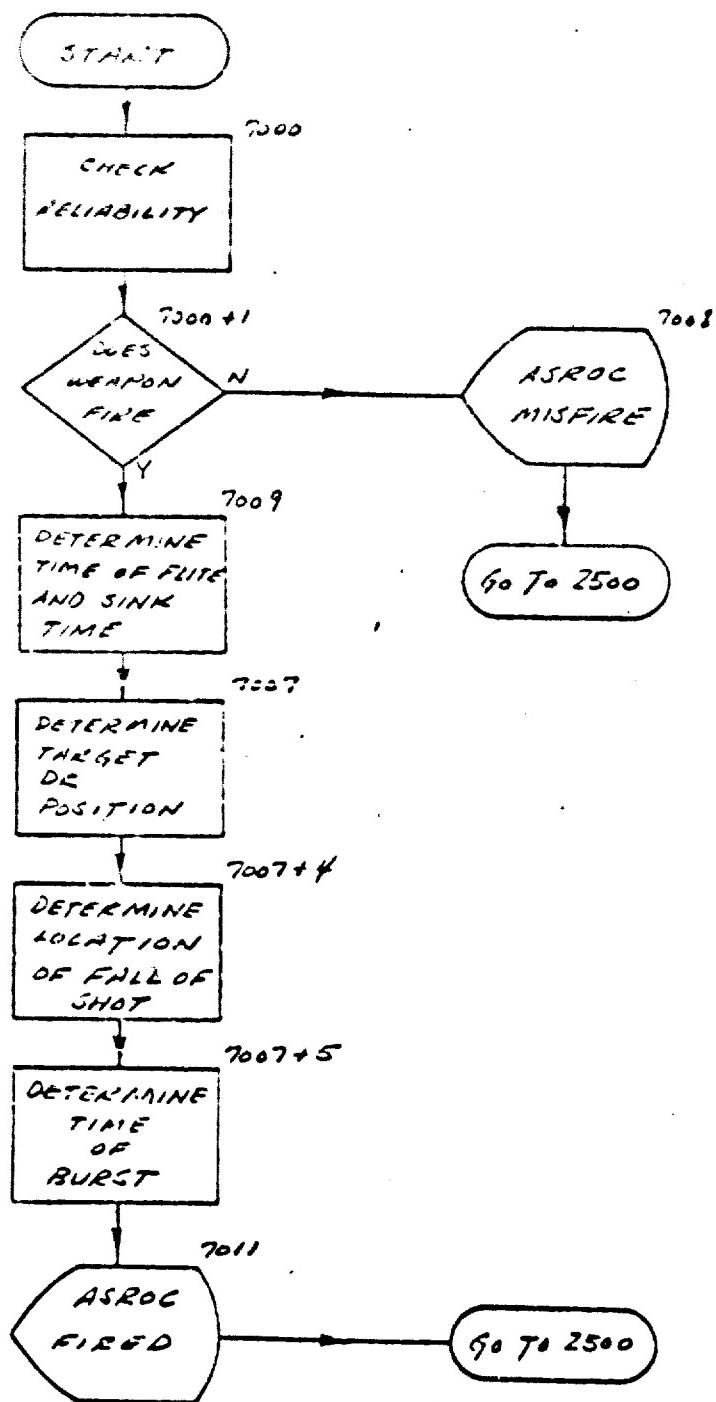


Reproduced From  
Best Available Copy

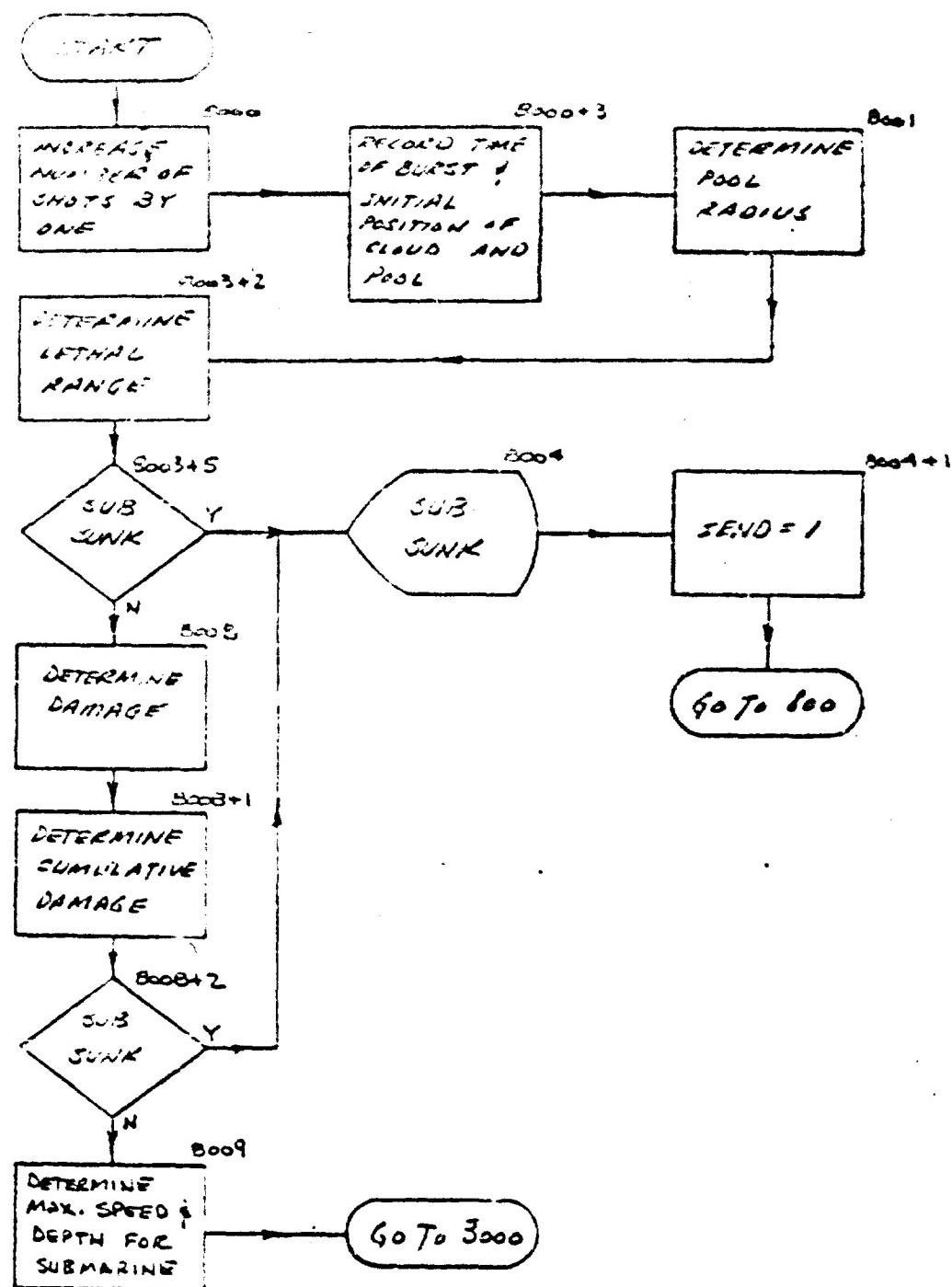
# CONTACT TRACKING MODEL



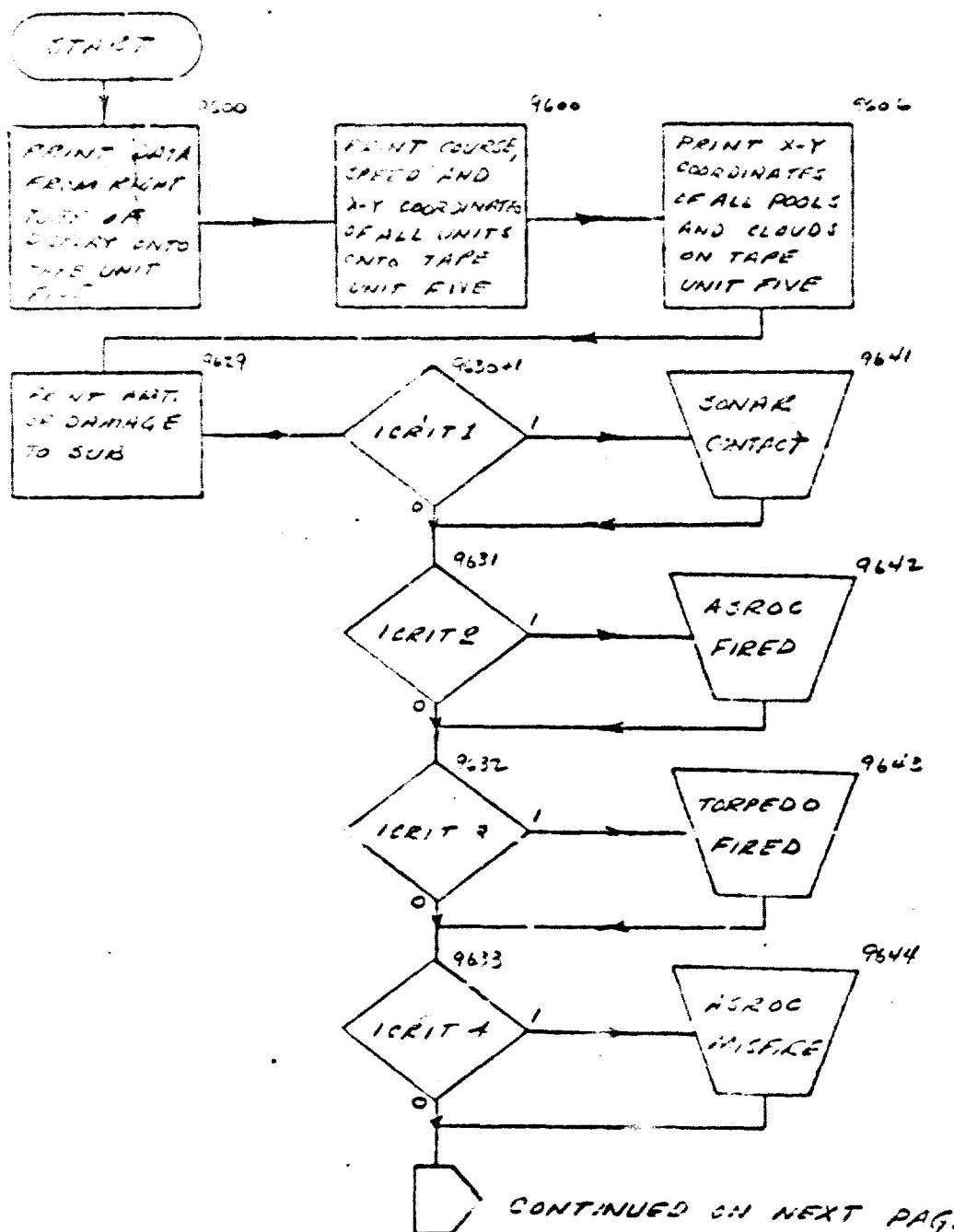
# WEAPON FIRING MODEL

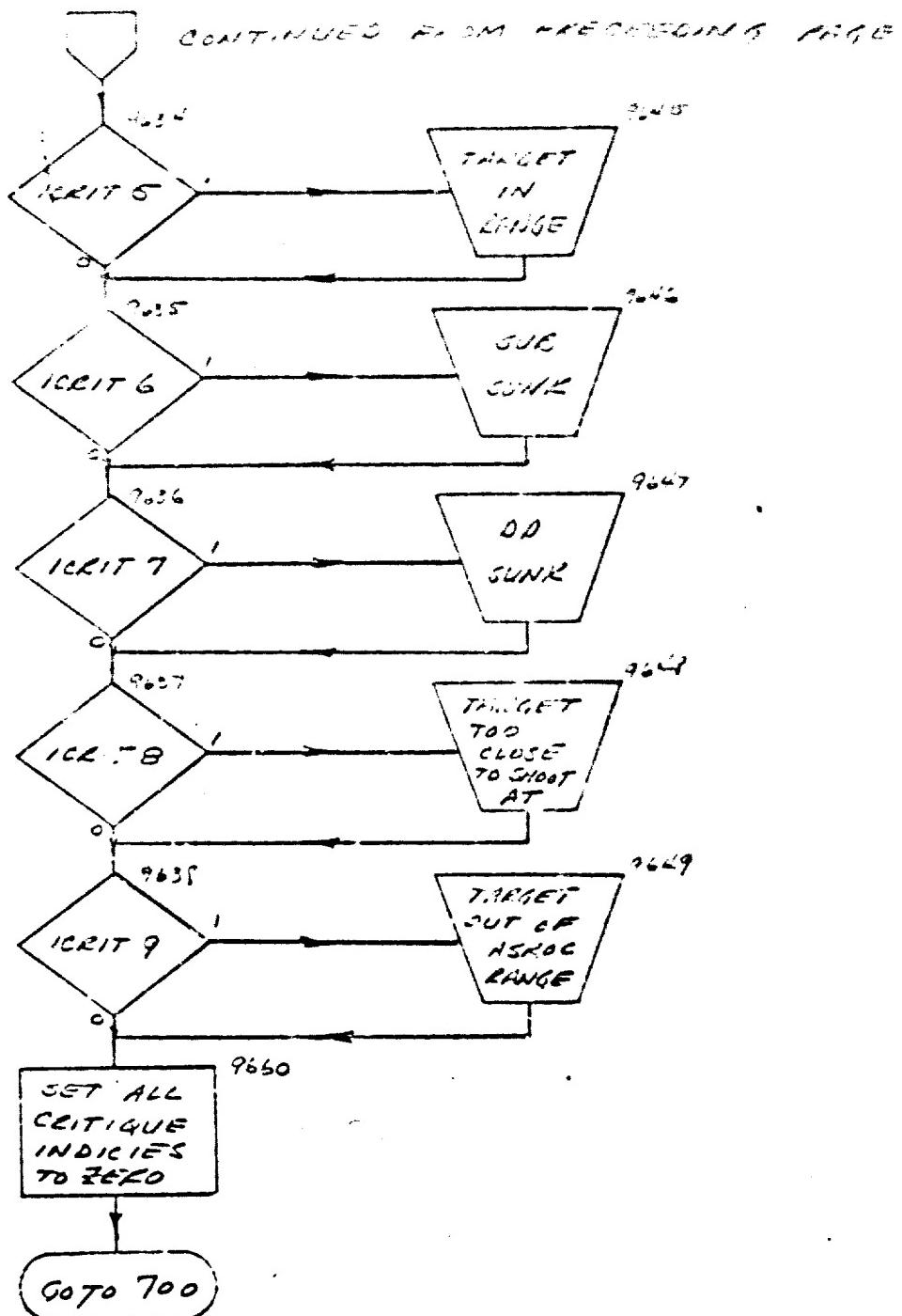


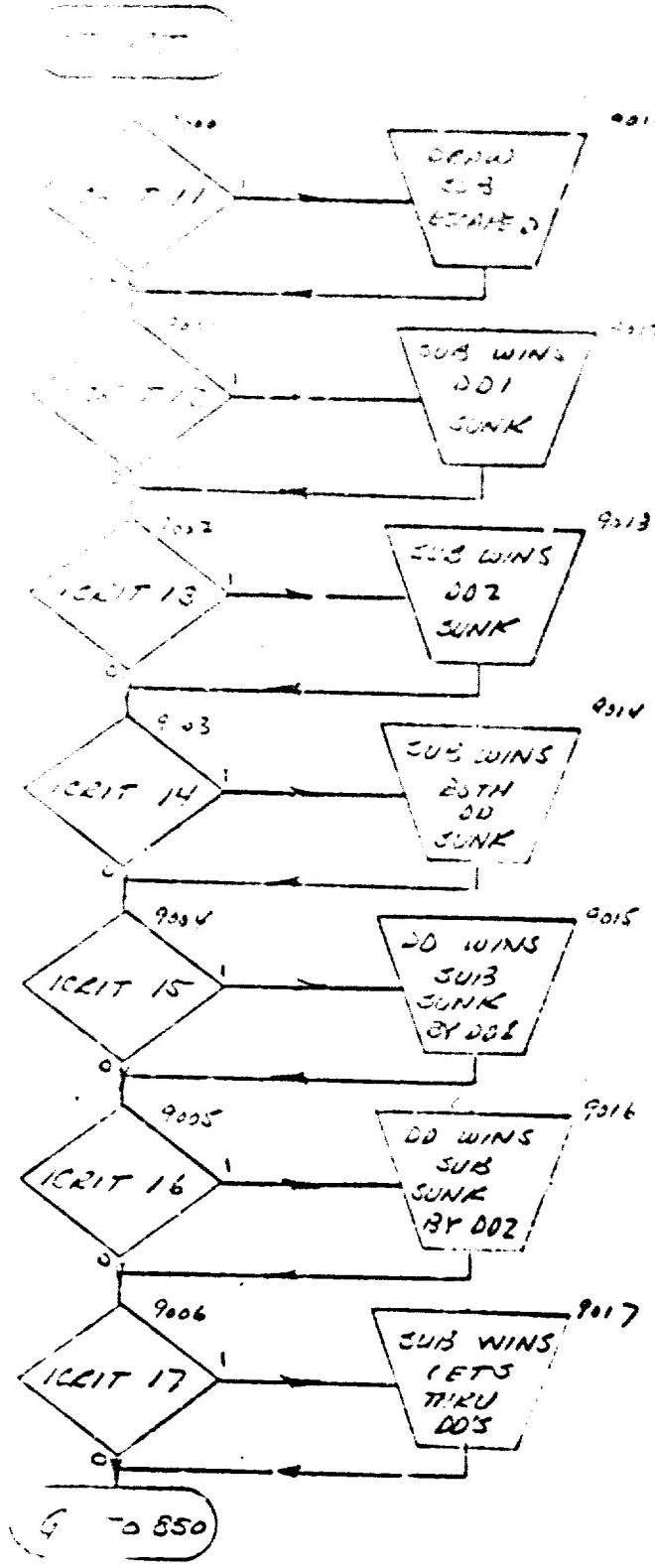
# EVALUATION MODEL



# CRITIQUE I







Reproduced From  
Best Available Copy

## APPENDIX III

### SUBROUTINES AND CDC 160 EXECUTIVE ROUTINE

This Appendix contains an explanation of the subroutines used in the main program and the CDC 160 executive routine used to connect the CDC 1604 with the DD 65 display. A listing of these subroutines can be found in Appendix IV.

#### SUBROUTINE RANVAR

Subroutine RANVAR is used as the basic random number generator. It generates random floating point numbers in the interval zero to one that are distributed uniformly in that interval. The random variables used in the main program are called from either UNIFORM, NORMAL or ERROR subroutines which in turn call RANVAR for input. The generator is a simple fixed point division utilizing the remainder from the Q register as the random number. This number is then mapped into the zero to one interval. Only one input is required to initialize this generator, namely IRANDOM.

#### SUBROUTINE UNIFORM

Subroutine UNIFORM is a three argument subroutine used to generate uniformly distributed random numbers in any interval. The arguments are:

1. The center of the interval.
2. The half width of the interval.
3. The output random number.

An example would be UNIFORM (5.0, 2.0, SUBS). This call would yield the variable SUBS, submarine speed, uniformly distributed in the interval with center at five knots and plus or minus two knots.

### SUBROUTINE NORMAL

Subroutine NORMAL is a three argument subroutine used to generate normally distributed random numbers with any mean and standard deviation. This normal distribution is generated by means of the sum of identically distributed (uniform) random variables. Twelve uniform numbers are used because:

1. The truncation is not significant.
2. Twelve reduces the formula to a summation and no division is required.

The arguments to this subroutine are:

1. The mean of the distribution.
2. The sigma of the distribution.
3. The output random number.

### SUBROUTINE ERROR

Subroutine ERROR is a five argument subroutine used to generate circular normal distributed random variables. This subroutine is used to determine the true fall of shot given the aiming point and CEP or sigma of the distribution of fall or shot. The arguments are:

1. The X-coordinate of the aim point.
2. The Y-coordinate of the aim point.
3. The sigma of the circular normal distribution.
4. The X-coordinate of the fall of shot.
5. The Y-coordinate of the fall of shot.

### SUBROUTINE CIRCLE

Subroutine CIRCLE is used to generate a series of points, 24 in number, every 15 degrees around the perimeter of a circle of predetermined

center and radius. The circle is then used in the formation of the circular clouds and pools of radiation displayed to the destroyer team. The arguments are:

1. The X-coordinate of the center of the circle with respect to the main coordinate system.
2. The Y-coordinate of the center of the circle with respect to the main coordinate system.
3. The radius of the circle with respect to the main coordinate system.
4. The X-coordinate of the center of the display with respect to the main coordinate system.
5. The Y-coordinate of the center of the display with respect to the main coordinate system.
6. The radius of the displayed area.
7. A vector of X-coordinates of the 24 points in the circle with respect to the display coordinate system.
8. A vector of Y-coordinates of the 24 points in the circle with respect to the display coordinate system.

#### SUBROUTINE DCIRCLE

DCIRCLE is a subroutine used to transmit the circle coordinates, generated in subroutine CIRCLE, to the CDC 160 from the CDC 1604. The arguments to this subroutine are:

1. ITRENO - the track of circle number by which this particular circle can be designated.
2. CHAR - a single letter or number in Hollerith form that is to be displayed as one of the points in the circle.
3. In the program MULNUC1 the letter c and p are used to distinguish between clouds and pools of radiation.

4. NUMPTS - the number of points in the circle.
5. IX - a vector of X-coordinates for the circle.
6. IY - a vector of Y-coordinates for the circle.

#### SUBROUTINE DTRACK

Subroutine DTRACK is used to transmit track data to the CDC 160 from the CDC 1604. This subroutine will send tracks of up to eight points to the CDC 160. The arguments to this subroutine are the same as those in DCIRCLE, in fact the two routines are identical with the exception of the allowable number of points.

#### SUBROUTINE DSTATUS

DSTATUS is a subroutine used to transmit information to the windows described in the section on Man Machine Interface. This information can be in two forms:

1. Program variables in fixed point form.
2. Messages in hollerith form.

The arguments are:

1. ITYPE - zero represents a numerical program variable is to be sent, while one indicates that an eight hollerith character word is to be sent.
2. NWIND - the number of the window to which the variable or word is to be sent.
3. IW - the field width of the variable.
4. INAME - the name of the variable to be transmitted.
5. IX - the X-coordinate of the lower left hand corner of the window.
6. IY - the Y-coordinate of the lower left hand corner of the window.

It should be noted that windows are 128 display coordinates units long. If a message in the form of words is to be sent to the display and it is longer than eight letters (the length of one window) it can be sent by means of more than one window. These windows should be displaced by 128 units in the X direction, thus the windows may form a continuous word of more than eight characters.

#### SUBROUTINE PARAMS

PARAMS is an eight argument subroutine used to query the CDC 160 as to the contents of eight selected windows. This subroutine allows the main program, in the CDC 16C4, to enter changes that have been made to the windows of the display by the player. This is the only method the player has of communicating with the program without interrupting the play. The eight arguments to this subroutine are:

- 1A - the contents of window 5
- 2A - the contents of window 6
- 3A - the contents of window 7
- 4A - the contents of window 8
- 5A - the contents of window 9
- 6A - the contents of window 9
- 7A - the contents of window 16
- 8A - the numeric value of a location in the CDC 160 that is controlled by the DD1 FIRE and DD2 FIRE buttons.

APPENDIX IV  
PROGRAM LISTING

This Appendix contains the computer program listing of the simulation MELNECI. The program blocks are in numerical order and the logic can be followed by cross referencing this Appendix with Appendix I. In this program all classified input parameters have been assigned fictitious values so that the program, as presented in this thesis, could remain unclassified.

••JOB#444F-MCMICHAEL,D.L.  
PROGRAM MULNUCI

MUL00010  
MUL00020

C C IN RUNNING THIS PROGRAM THE PLAYER WILL HAVE THE OPTION  
C C OF CHANGING THE FOLLOWING INPUT PARAMETERS.

YIELD	SIZE OF ASROC WARHEAD
AROCMAX	ASROC MAXIMUM RANGE
DOB	DEPTH OF BURST
TFACTOR	1 - REAL TIME 2 - DOUBLE TIME
GRAD	WATER TEMPERATURE GRADIENT
DDSMAX(1)	DD1 MAXIMUM SPEED AVAILABLE
DOSMAX(2)	DD2 MAXIMUM SPEED AVAILABLE
SUBSMAX	SUBS MAXIMUM SPEED AVAILABLE
RANDOM	A RANDOM NUMBER TO START GAME
STRESS	YIELD STRESS OF THE SUBMARINE HULL
HULL	SUBMARINE HULL THICKNESS
DDX(1)	DD1 X-COORDINATE
DDX(2)	DD2 X-COORDINATE
DDY(1)	DD1 Y-COORDINATE
DDY(2)	DD2 Y-COORDINATE

C 1SUB 0 - COMPUTER CONTROL 1 - SUB TEAM CONTROL  
C INITIAL 0 - SUB AT ORIGIN 1 - SUB POSIT RANDOM  
C ISTRAT 0 - RUN 1 - UP MIDDLE 2 - END RUN  
C THESE PARAMETERS ARE SET TO NOMINAL INITIAL VALUES AND  
C WILL BE DISPLAYED AS INPUT PARAMETERS. IF A CHANGE OF THESE  
C VALUES IS DESIRED IT CAN BE MADE AT THE TIME THE SCREEN  
C DISPLAYS (VARIABLES ARE) AND A LIST OF VARIABLES. AT THIS  
C TIME THE N-TH VARIABLE IN THE LIST CAN BE CHANGED BY  
C TYPING A AND THE NUMBER N, AND THEN •, AND THE NEW VALUE OF  
C THE SELECTED PARAMETER. AS AN EXAMPLE IF THE THIRD  
C VARIABLE IS TO BE CANGED TO 3.1 THE PLAYER WOULD TYPE  
C A3=3.1 AND THEN DEPRESS THE OUTPUT BUTTON ON THE DD65.  
C THE COMPUTER WILL THEN RETURN (OK) IF THE PARAMETER IS OF  
C THE CORRECT FORM. AFTER ALL CHANGES HAVE BEEN ENTERED  
C THE PLAYER WILL TYPE END AND THE GAME WILL CONTINUE.  
C HAVING ENTERED ALL PARAMETER CHANGES THE GAME WILL  
C PROCEED AUTOMATICALLY. THE PLAYER MAY THEN MANEUVER THE  
C DESTROYERS BY THE PROCEDURE OUTLINED IN THE SECTION

C OF THE THESIS TITLED MAN MACHINE INTERFACE.

```
DIMENSION DDX(2),DDY(2),CONTR(2),CONTB(2),CONTc(2),CCNTS(2),  
1      IDOPLER(2),MARKS(2),X(2,5),Y(2,5),D(S(2)),  
2      NMARKS(2),DOB(2),CONTX(2),ONTY(2),SSB(2),DSMAX(2),  
3      ICLASS(2),IDDC(2),RADRATE(2),RADDSE(2),IS(L(2)),  
4      SB(2),ICONTC(2),ICONTB(2),TBURST(10),CLOUD(10),  
5      CLOUDX(10),CLOUDY(10),POOLX(10),POOLY(10),POCLR(10),  
6      DISTC(2,10),DISTP(2,10),TEMPC(2,10),TEMPF(2,10),  
7      IDDX(2),IDDY(2),IDDS(2),XDD(2,8),YDD(2,8),  
8      TXDD(2,2),TYDD(2,8),IXDD(2,2),IYDD(2,8),IX)D(8),2ZY(3),MULCO110  
9      IXDD2(8),IYDD1(8),IYDD2(8),XSUB(2,8),YSUB(2,8),Z2X(3),  
DIMENSION TXSUB(2,8),TYSUB(2,8),IXSUB(2,8),IYSUB(2,8),XSUB1(8),  
1      IYSUB1(8),IXSUB2(8),IYSUB3(8),IYSUB2(8),IRCX(24),IRCY(24),  
COMMON IRANDOM,RANDOM  
C INPUTS 100  
C 100 NRDD=2  
      TFACTOR=1.0  
      EXF(1000B)  
      DO 101 I=1,NRDD  
101  DS MAX(1)=3C.0  
      SUBSMAX=20.0  
      STRESS=50.0  
      HULL=2.50  
      SAFETY=.5  
      DDX(1)=-5000.0  
      DDY(1)=-20000.0  
      DDX(2)=5000.0  
      DDY(2)=-20000.0  
      DDC(1)=010.0  
      DDS(1)=15.0  
      DOC(2)=010.0  
      DOS(2)=15.0  
      YIELD=2.0  
      MULCO150  
      MULCO160  
      MULCO170  
      MULCO180  
      MULCO190  
      MULCO200  
      MULCO210  
      MULCO220  
      MULCO230  
      MULCO240  
      MULCO250  
      MULCO260  
      MULCO270  
      MULCO280  
      MULCO290  
      MULCO300  
      MULCO310  
      MULCO320  
      MULCO330  
      MULCO340  
      MULCO350  
      MULCO360
```

```

THERMO=120.0
AROMAX=8000.0
DOB=700.0
SR=18.0
IRANDOM=574523
NRANDOM=300
ISUB=0
K=0
KGRAF=0
CALL UNIFORM(.5,.5,TEMP)
IFI(TEMP--.5)102,102,103
102 INITIAL=0
GO TO 104
103 INITIAL=1
104 CONTINUE
CALL NORMAL(-.472,.096,GRAD)
IFI(GRAD+.37)106,105,105
105 GRAD=-.37
106 CONTINUE
CALL UNIFORM(.5,.5,TEMP)
IFI(TEMP--.3)108,108,107
107 IF(I(TEMP--.6)109,109,110
108 ISTRAT=0
GO TO 111
109 ISTRAT=1
GO TO 111
110 ISTRAT=2
111 CONTINUE
GO TO 200
C SET CONSTANTS 200
C 200 TSTEP=.30/0/TFACTOR
ISTEP=TSTEP
X0=100.0
Y0=100.0
MUL00370
MUL00350
MUL00330
MUL00460
MUL00410
MUL00420
MUL00430
MUL00440
MUL00450
MUL00460
MUL00470
MUL00480
MUL00490
MUL00500
MUL00510
MUL00520
MUL00530
MUL00540
MUL00550
MUL00560
MUL00570
MUL00580
MUL00590
MUL00600
MUL00610
MUL00620
MUL00630
MUL00640
MUL00650
MUL00660
MUL00670
MUL00680
MUL00690
MUL00700
MUL00710
MUL00720

```

```

R=25000.0
ILOGIC=0
DO 201 I=1,NRDD
RADRATE(I)=0.0
RADDSE(I)=0.0
ISOL(I)=0
ICLASS(I)=0
NMARKS(I)=0
201 MARKS(I)=0
NEXT=0
GTIME=0.0
NSHOTS=0
NOSHOOT=0
DAMAGE=0.0
DAMAGET=0.0
TOB=0.0
B=0.0
NCONBER=0
IFIRE=0
NPTS=0
IEND=0
NZERO=-0
IONE=1
GO TO 300
C INITIALIZE 300
C
300 DO 332 I=1,NRANDOM
332 CALL RANVAR
IF((INITIAL)330,330,331
330 SUBX=0.0
SUBY=0.0
SUBD=0.0
SUBC=180.0
SUBS=5.0
GO TO 333

```

```

331 CALL UNIFORM(500C,0.5000,0,SUBSY)
    CALL UNIFORM(0,10000,0,SUBA)
    SUBD=300.0
    SUBC=100.0
    SUPS=5.0
333 CONTINUE
    CRUSH=1.5*32*MULC1*STRESS
    SUBDMAX=SAFETY*CRUSd
    CALL UNIFORM(180.0,179.9,WINDD)
    IWINDD=WINDD
    CALL UNIFORM(15.0,15.0,WINDV)
    AROCMIN=1470.0*YIELD**333
    ESR=33.3*SQRTFL(-300.0/(GRAD**05+0.01E1))
    IF(WINDV<=3.0)301,301,302
301 ISS=1
    GO TO 313
302 IF(WINDV>=6.0)303,303,304
303 ISS=2
    GO TO 313
304 IF(WINDV>=10.0)305,305,306
305 ISS=3
    GO TO 313
306 IF(WINDV>=16.0)307,307,308
307 ISS=4
    GO TO 313
308 IF(WINDV>=21.0)309,309,310
309 ISS=5
    GO TO 313
310 IF(WINDV>=27.0)311,311,312
311 ISS=6
    GO TO 313
312 ISS=7
    GO TO 313
313 CONTINUE
    TEMP1=YIELD**333
    TEMP2=YIELD**25
    IF(DCB>75.0)TEMP1,316,314,314

```



```

DO 339 J=1,8
XSUB(1,J)=-0.0
339 YSUB(1,J)=-0.0
340 IF(IISUB)343,343,341
341 WRITE OUTPUT TAPE 9,342,NRDD,SUBSMAX,STRESS,HULL,SAFETY,THERMO.
1TFACOR,GRAD,SUBDMAX
342 FORMAT(6X,16HINPUT PARAMETERS/6X,24HNNUMBER OF DESTROYERS •••,13/
16X,24HMAXIMUM SUB SPEED •••.,F5.1,2X,5HKNOTS/6X,24HHULL YIELD SIMUL01880
PRESS •••.,F5.1,2X,4HKPSI/6X,24HHULL THICKNESS •••.,F5.7,2X,MUL01970
36HINCHES/6X,24HDEPTH OF THERMOCLINE •••.,F5.1,4H FT/
46X,24HTIME FACTOR •••.,F5.1/6X,24HWATER TEMP GRADIENT •••.MUL01910
5,F6,3,10H DEG/100FT/6X,24HMAXIMUM SUB DEPTH •••.,F6,1.5H FEET) MUL01920
MUL01930
343 CONTINUE
344 PRINT 345,NRDD,YIELD,THERMO,AROCMAX,DOB,SR,TFACTOR,GRAD
345 FORMAT(6X,16HINPUT PARAMETERS/6X,24HNNUMBER OF DESTROYERS •••,13/
16X,24HSIZE OF ASROC WARHEAD •••.,F5.1,4H KT/6X,24HDEPTH OF THERMOCLMUL01950
2INE •••.,F5.1,4H FT/6X,24HMAXIMUM RANGE OF A SROC •••.,F7.1,6H YARDS/ MUL01960
36X,24HDEPTH OF BURST •••.,F7.1,3H FT/6X,24HSINK RATE OF WARMEMUL01930
4AD •••.,F5.1,9H FT/SEC/6X,24HTIME FACTOR •••.,F5.1/6X,24HMUL01930
5WATER TEMP GRADIENT •••.,F6,3,10H DEG/100FT// ////)
MUL02000
MUL02010
GO TO 950
MUL02020
MUL02030
MUL02040
MUL02050
MUL02060
MUL02070
MUL02080
MUL02090
MUL02100
MUL02110
MUL02120
MUL02130
MUL02140
MUL02150
MUL02160
C PLOT POSITIONS 400
C
400 DO 401 I=1,ONE,NRDD
    IDDX(1)=DDX(1)/100.0
    IDDY(1)=DDY(1)/100.0
    IDDC(1)=DDC(1)
401 IDDS(1)=DDS(1)
    IX0=X0/100.0
    IY0=Y0/100.0
    IR=R/1000.0
    CALL DSTATUS(0,1,8,IDDX(1),-200,200)
    CALL DSTATUS(0,2,8,IDDY(2),136,200)
    CALL DSTATUS(0,3,8,IDDY(1),-200,176)
    CALL DSTATUS(0,4,8,IDDY(2),136,176)

```

```

CALL DSTATUS(0,5,3,IDDG(1),-200,-176)
CALL DSTATUS(0,6,2,IDDG(1),-200,-200)
CALL DSTATUS(0,7,3,IDDG(2),136,-176)
CALL DSTATUS(0,8,2,IDDG(2),136,-200)
CALL DSTATUS(0,9,8,IX0,-32,200)
CALL DSTATUS(0,10,8,IY0,-32,176)
CALL DSTATUS(0,11,8,NZERO,-254,0)
CALL DSTATUS(0,12,8,NZERO,-126,0)
CALL DSTATUS(0,13,8,NZERO,2,0)
CALL DSTATUS(0,14,8,NZERO,130,0)
CALL DSTATUS(0,15,8,NZERO,-32,-176)
CALL DSTATUS(0,16,8,IR,-32,-200)
IF(NPTS-8)402,433,434

402 NPTS=NPTS+1
GO TO 433

434 NPTS=8
433 DO 404 I=1,ONE,NRDO
    DO 403 J=1,7
        XDD(1,9-J)=XDD(1,8-J)
        YDD(1,9-J)=YDD(1,8-J)
        XDD(1,1)=DDX(1)
        YDD(1,1)=DDY(1)
    DO 406 I=1,ONE,NRDO
        DO 408 J=1,NPTS
            TXDD(1,J)=XDD(1,J)-XO
            TYDD(1,J)=YDD(1,J)-YO
            IXDD(1,J)=TXDD(1,J)*255.0/R
            IYDD(1,J)=TYDD(1,J)*255.0/R
            IF(XABSF(IXDD(1,J))-255)>06,406,405
    405 IXDD(1,J)--0
    406 IF(XABSF(IYDD(1,J))-255)>08,408,407
    407 IYDD(1,J)--0
    408 CONTINUE
    DO 409 I=1,NPTS
        IXDD1(1)=IXDD(1,1)
        IXDD2(1)=IXDD(2,1)

```

```

IYDD1(1)=IYDD(1,1)
IYDD2(1)=IYDD(2,1)
IT=8H1
CALL DTRACK(1,IT,NPTS,IXDD1,IYDD1)
IT=8H2
CALL DTRACK(2,IT,NPTS,IXDD2,IYDD2)
410 IF(INMARKS(1))411,411,412
411 NA=2
GO TO 413
412 NA=1
413 IF(INMARKS(2))414,414,415
414 NB=1
GO TO 416
415 NB=2
416 IF(NB-NA)425,417,417
417 DO 419 I=NA,NB
DO 418 J=1,7
XSUB(I,9-J)=XSUB(I,8-J)
418 YSUB(I,9-J)=YSUB(I,8-J)
XSUB(I,1)=CONTX(I)
419 YSUB(I,1)=CONTY(I)
DO 423 I=NA,NB
DO 423 J=1,8
TXSUB(I,J)=XSUB(I,J)-X0
TYSUB(I,J)=YSUB(I,J)-Y0
IXSUB(I,J)=TXSUB(I,J)*255.0/R
IYSUB(I,J)=TYSUB(I,J)*255.0/R
IF(XABSF(IXSUB(I,J))-255)421,421,420
420 IXSUB(I,J)=0
421 IF(XABSF(IYSUB(I,J))-255)423,423,422
422 IYSUB(I,J)=0
423 CONTINUE
DO 424 I=1,8
IXSUB(I)=IXSUB(I,1)
IYSUB(I)=IYSUB(I,1)
DO 432 I=1,8

```

```

IXSUB2(1)=IXSUB(2,1)          MUL02890
IYSUB2(1)=IYSUB(2,1)          MUL02900
432 GO TO 427                 MUL02910
425 DO 426 I=1,8               MUL02920
IXSUB1(1)=0                    MUL02930
IYSUB1(1)=0                    MUL02940
IXSUB2(1)=0                    MUL02950
IYSUB2(1)=0                    MUL02960
426 IT=8HX                     MUL02970
CALL DTRACK(3,IT,0,IXSUB1,IYSUB1)
IT=8HX                         MUL02980
CALL DTRACK(4,IT,0,IXSUB2,IYSUB2)
CONTINUE
IF(NSHOTS)431,431,428
428 DO 429 I=1,NSHOTS
CALL CIRCLE(CLOUDX(1),CLOUDY(1),CLOUDR(1),X0,Y0,R,ICIRCX,ICIRCY)
IT=8HC
429 CALL DCIRCLE(1,IT,24,ICIRCX,ICIRCY)
DO 430 I=1,NSHOTS
CALL CIRCLE(PPOOLX(1),POOLY(1),POOLR(1),X0,Y0,R,ICIRCX,ICIRCY)
IPLUS=1+NSHOTS
IT=8HP
430 CALL DCIRCLE(IPLUS,IT,24,ICIRCX,ICIRCY)
431 CONTINUE
GO TO 500

C DISPLAY DATA 500
C
500 IF(MARKS(1)) 502,502,504
502 IF(MARKS(2)) 511,511,503
503 IF(MARKS(2)-3)512,513,513
504 IF(MARKS(2)) 505,505,506
505 IF(MARKS(1)-3) 515,514,514
506 IF(MARKS(1)-3)508,507,507
507 IF(MARKS(2)-3)517,516,516
508 IF(MARKS(2)-3)519,518,518

```

```

511 PRINT 521,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX(1),I=1,NRDD),
1(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD)
GO TO 599

512 PRINT 522,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX(1),I=1,NRDD),
1(CLASS(2),CONTR(2),ICONTB(2),IDOPLER(2),
2(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD)
GO TO 599

513 PRINT 523,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX(1),I=1,NRDD),
1(CLASS(2),CONTR(2),ICONTB(2),IDOPLER(2),
2ICONTC(2),CONTS(2),ISOL(2),
3(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD)
GO TO 599

514 PRINT 524,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX(1),I=1,NRDD),
1(CLASS(1),CONTR(1),ICONTB(1),IDOPLER(1),
2ICONTC(1),CONTS(1),ISOL(1),
3(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD)
GO TO 599

515 PRINT 525,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX(1),I=1,NRDD),
1(CLASS(1),CONTR(1),ICONTB(1),IDOPLER(1),
2(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD)
GO TO 599

516 PRINT 526,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX(1),I=1,NRDD),
1(CLASS(1),I=1,NRDD),(CONTR(1),I=1,NRDD),(ICONTB(1),I=1,NRDD),
2(IDOPLER(1),I=1,NRDD),
3(ICONTC(1),I=1,NRDD),(CONTS(1),I=1,NRDD),(ISOL(1),I=1,NRDD),
4(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD)
GO TO 599

517 PRINT 527,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX(1),I=1,NRDD),
1(CLASS(1),I=1,NRDD),(CONTR(1),I=1,NRDD),(ICONTB(1),I=1,NRDD),
2(IDOPLER(1),I=1,NRDD),
3ICONTC(1),CONTS(1),ISOL(1),
4(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD)
GO TO 599

518 PRINT 528,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX(1),I=1,NRDD),
1(CLASS(1),I=1,NRDD),(CONTR(1),I=1,NRDD),(ICONTB(1),I=1,NRDD),
2(IDOPLER(1),I=1,NRDD),

```

```

31CONT(2),CONT(2),ISOL(2),
4(RADRATE(1),I=1,NRDD),RADDOSE(1),I=1,NRDD)
GO TO 599
MUL03610
MUL03620
MUL03630
MUL03640
MUL03650
MUL03660
MUL03670
MUL03680
MUL03690
MUL03700
MUL03710
MUL03720
MUL03730
MUL03740
MUL03750
MUL03760
MUL03770
MUL03780
MUL03790
MUL03800
MUL03810
MUL03820
MUL03830
MUL03840
MUL03850
MUL03860
MUL03870
MUL03880
MUL03890
MUL03900
MUL03910
MUL03920
MUL03930
MUL03940
MUL03950
MUL03960

519 PRINT 529,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX()),I=1,NRDD),
1((ICLASS(1),I=1,NRDD),(CONTR(1),I=1,NRDD),(ICONTB(1),I=1,NRDD),
2(DOPPLER(1),I=1,NRDD),
3(RADRATE(1),I=1,NRDD),RADDOSE(1),I=1,NRDD)
GO TO 599
MUL03610
MUL03620
MUL03630
MUL03640
MUL03650
MUL03660
MUL03670
MUL03680
MUL03690
MUL03700
MUL03710
MUL03720
MUL03730
MUL03740
MUL03750
MUL03760
MUL03770
MUL03780
MUL03790
MUL03800
MUL03810
MUL03820
MUL03830
MUL03840
MUL03850
MUL03860
MUL03870
MUL03880
MUL03890
MUL03900
MUL03910
MUL03920
MUL03930
MUL03940
MUL03950
MUL03960

521 FORMAT 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X,
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X,
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL03710
35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/
45X,13HCONTACT CLASS/5X,11HSOAR RANGE/5X,13HSOAR BEARING/5X,
57HDOPPLER/5X,13HTARGET COURSE/5X,12HTARGET SPEED/5X,15HFIRING SOLUTION/MUL03740
6TION/5X,14HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,
75X,F5.0,6X,F5.0//)
522 FORMAT 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X,
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X,
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL03790
35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/
45X,13HCONTACT CLASS,20X,11/5X,11HSOAR RANGE,18X,F6.0/5X,13HSOAR
5BEARING,18X,13/5X,7HDOPPLER,26X,11/5X,13HTARGET COURSE/5X,
612HTARGET SPEED/5X,15HFIRING SOLUTION/5X,
714HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0,
86X,F5.0//)
523 FORMAT 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X,
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X,
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL03860
35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/
45X,13HCONTACT CLASS,20X,11/5X,11HSOAR RANGE,18X,F6.0/5X,13HSOAR
5BEARING,18X,13/5X,7HDOPPLER,26X,11/5X,13HTARGET COURSE,18X,13/5X,
612HTARGET SPEED,20X,F3.0/5X,15HFIRING SOLUTION,18X,11/5X,
714HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0,
86X,F5.0//)
524 FORMAT 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X,
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X,

```

212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11//25X,3HDD1,8X,3HDD2 / MUL03970  
 35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/  
 45X,13HCONTACT CLASS,9X,11/5X,11HSONAR RANGE,7X,F6.0/5X,13HS0NAR MUL03980  
 5BEARING,7X,13/5X,7HDOPPLER,15X,11/5X,13HTARGET COURSE,7X,13/5X,  
 612HTARGET SPEED,9X,F3.0/5X,15HFIRING SOLUTION,7X,11/5X,  
 714HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0.  
 86X,F5.0//)

525 FORMAT( 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL04040  
 113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/,X, MUL04050  
 212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11//25X,3HDD1,8X,3HDD2 / MUL04060  
 35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/  
 45X,13HCONTACT CLASS,9X,11/5X,11HS0NAR RANGE,7X,F6.0/5X,13HS0NAR MUL04070  
 5BEARING,7X,13/5X,7HDOPPLER,15X,11/5X,13HTARGET COURSE,5X,  
 612HTARGET SPEED/5X,15HFIRING SOLUTION/5X,  
 714HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0.,  
 86X,F5.0// )

526 FORMAT( 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL04120  
 113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL04130  
 212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11//25X,3HDD1,8X,3HDD2 / MUL04140  
 35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/  
 45X,13HCONTACT CLASS,9X,11,10X,11/5X,11HS0NAR RANGE,7X,F6.0,5X,F6.0/MUL04150  
 5/5X,13HS0NAR BEARING,7X,13,8X,13/5X,7HDOPPLER,15X,11,10X,11/, MUL04160  
 65X,13HTARGET COURSE,7X,13,8X,13/5X,12HTARGET SPEED,9X,F3.0,8X,F3.0/MUL04170  
 7/5X,15HFIRING SOLUTION,7X,11,10X,11/5X,  
 814HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0.,  
 96X,F5.0// )

527 FORMAT( 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL04230  
 113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL04240  
 212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11//25X,3HDD1,8X,3HDD2 / MUL04250  
 35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/  
 45X,13HCONTACT CLASS,9X,11,10X,11/5X,11HS0NAR RANGE,7X,F6.0,5X,F6.0/MUL04270  
 5/5X,13HS0NAR BEARING,7X,13,8X,13/5X,7HDOPPLER,15X,11,10X,11/, MUL04280  
 65X,13HTARGET COURSE,7X,13,5X,12HTARGET SPEED,9X,F3.0  
 7/5X,15HFIRING SOLUTION,7X,11,5X,  
 814HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0.,  
 96X,F5.0// )

```

528. FORMAT( 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X,
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X,
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11//25X,3HDD1,8X,3HDD2/MUL04350
MUL04340
35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/
MUL04360
45X,13HCONTACT CLASS,9X,11/5X,11HSONAR RANGE,7X,F6.0,5X,F6.0MUL04370
MUL04380
5/5X,13HSONAR BEARING,7X,13,8X,13/5X,7HDOPPLER,15X,11,10X,11/
MUL04390
65X,13HTARGET COURSE,18X,13/5X,12HTARGET SPEED,20X,F3.0
MUL04400
7/5X,15HFIRING SOLUTION,18X,11/5X,
MUL04410
814HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0,
MUL04420
96X,F5.0//) MUL04420

529 FORMAT( 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X,
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X,
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11//25X,3HDD1,8X,3HDD2/MUL04450
MUL04460
35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F3.0/
MUL04470
45X,13HCONTACT CLASS,9X,11,10X,11/5X,11HSONAR RANGE,7X,F6.0,5X,F6.0MUL04480
MUL04490
5/5X,13HSONAR BEARING,7X,13,8X,13/5X,7HDOPPLER,15X,11,10X,11/
MUL04490
65X,13HTARGET COURSE/5X,12HTARGET SPEED/5X,15HFIRING SOLUTION/5X,
MUL04500
714HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0,
MUL04510
86X,F5.0//) MUL04520
MUL04530
MUL04540
MUL04550
MUL04560
MUL04570
MUL04580
MUL04590
MUL04600
MUL04610
MUL04620
MUL04630
MUL04640
MUL04650
MUL04660
MUL04670
MUL04680

C TIME LOOP 700
C
700 LD,(5006B),STA(DUMMY),LDAT(DUMMY),STA(ICLOCK)
    IF(ICLOCK-NEXT)701,702,702
701 GO TO 700
702 NEXT=ICLOCK+1 STEP
    GTIME=GTIME+.5
    GO TO 800
C
C GAME OVER 800
C
800 IF(IEND)801,801,802
801 GO TO 4000
802 GO TO 9000
C

```

C ENTER CHANGES 900

C 900 CALL PARAMS(IDDC(1), IDDS(1), IDDC(2), IDDS(2), IX0, IY0, IR, M)

DO 914 I=1,2  
IF(IDDC(1)-360)902,901,901

901 IDDC(1)=IDDC(1)-360  
902 IF(IDDC(1)903,914,914

903 IDDC(1)=IDDC(1)+360  
914 DDC(1)=IDDC(1)

DO 907 I=10NE,NRDD  
IF(IDDS(1)1904,905,905

904 IDDS(1)=0  
905 DDS(1)=IDDS(1)

IF(IDDS(1)-DDSMAX(1))907,907,906  
906 DDS(1)=DDSMAX(1)  
907 CONTINUE

X0=100\*IX0  
Y0=100\*IY0

R=1000\*IR  
IF(M-1)908,909,910

908 ISHOOT1=0  
ISHOOT2=0  
GO TO 913

909 ISHOOT1=1  
ISHOOT2=0  
GO TO 913

910 IF(M-3)911,912,912  
911 ISHOOT1=0  
ISHOOT2=1  
GO TO 913

912 ISHOOT1=1  
ISHOOT2=1  
913 CONTINUE  
GO TO 400

C ENTER INPUT CHANGES 950

MUL04690  
MUL04700  
MUL04710  
MUL04720  
MUL04730  
MUL04740  
MUL04750  
MUL04760  
MUL04770  
MUL04780  
MUL04790  
MUL04800  
MUL04810  
MUL04820  
MUL04830  
MUL04840  
MUL04850  
MUL04860  
MUL04870  
MUL04880  
MUL04890  
MUL04900  
MUL04910  
MUL04920  
MUL04930  
MUL04940  
MUL04950  
MUL04960  
MUL04970  
MUL04980  
MUL04990  
MUL05000  
MUL05010  
MUL05020  
MUL05030  
MUL05040

```

C      950 IF(K-1)951,400,400
951 K=1
      PRINT 952
952 FORMAT(4OHVARIABLES ARE YIELD,AROCMAX,DOB,TFACTOR,/4OHGRAD,DDSMAX(MUL05050
11),DDSMAX(2),SUBSMAX,NRANDOM)
      CALL CHANGE(YIELD,AROCMAX,DOB,TFACTOR,GRAD,DDSMAX(1),DDSMAX(2)).
1SUBSMAX,NRANDOM)
      PRINT 953
953 FORMAT(4OHVARIABLES ARE STRESS,HULL,DDX(1),DDY(1),/33HDX(2),DDY(2MUL05140
1),ISUB,INITIAL,ISTRAT)
      CALL CHANGE(STRESS,HULL,DDX(1),DDY(1),DDX(2),DDY(2),ISUB,INITIAL,
1ISTRAT)
      955 GO TO 200
C      PLOT GENERATOR MODEL 1000
C
1000 DO 1001 I=1,ONE,NRDD
      DDX(1)=DDX(1)+DDS(1)*16.67*SINF(DDC(1)/57.295)
1001 DDY(1)=DDY(1)+DDS(1)*16.67*COSF(DDC(1)/57.295)
      SUBX=SUBX+SUBS*16.67*SINF(SUBC/57.295)
      SUBY=SUBY+SUBS*16.67*COSF(SUBC/57.295)
      IF(INSHOTS)1015,1015,1002
1002 IF(IDEEP-2)1004,1006,1003
1003 IF(IDEEP-4)1008,1010,1010
1004 DO 1005 I=1,NSHOTS
1005 CLOUDR(1)=237.0*YIELD**.333*(5.85*LOGF((GTIME-TBURST(1))/
110.4450*YIELD**.1671+.731+.802)
      GO TO 1012
1006 DO 1007 I=1,NSHOTS
1007 CLOUDR(1)=125.0*DOB**.167*YIELD**.278*(5.85*LOGF((GTIME-TBURST
1(1))/.0.3240*YIELD**.944*DOB**.1671+.731+.802)
      GO TO 1012
1008 DO 1009 I=1,NSHOTS
1009 CLOUDR(1)=500.0*(YIELD/(DOB+33.0))**.333*(16.7*LOGF((GTIME-TBURST
1(1))/.0.6450*(YIELD/(DOB+33.0))**.1671+.4.54)

```

```

60 TO 1012          MUL0541C
1010 DO 1011 I=1,NSHOTS   MUL0542C
1011 CLOUDR(1)=500.0*(YIELD/(DOB+33.0))**.333*(16.7*LOGF((GTIME-TBURST
1011)/(0.6450*(YIELD/(DOB+33.0))**.167)-1.0)+4.54)   MUL0543C
1012 CONTINUE      MUL0544C
1013 DO 1014 I=1,NSHOTS   MUL0545C
    CLOUDX(1)=CLOUDX(1)-WINDV*16.67*SINF(WINDD/57.295)   MUL0547C
    CLOUDY(1)=CLOUDY(1)-WINDV*i6.67*COSF(WINDD/57.295)   MUL0548C
1014 GO TO 2000      MUL0549C
C INTERACTIONS 2000
C
2000 I=1           MUL0550C
    GO TO 5000          MUL0551C
2100 IF(NOSHOOT)Z102,Z102,Z101          MUL0552C
2101 GO TO 2500          MUL0553C
2102 GO TO 2400          MUL0554C
2200 IF(INKMARKS=1)Z2201,Z2201,Z203          MUL0555C
2201 INAME=8H SONAR          MUL0556C
    CALL DSTATUS(1,12,8,INAME,-126,0)          MUL0557C
    INAME=8H          MUL0558C
    CALL DSTATUS(1,11,8,INAME,-254,0)          MUL0559C
    CALL DSTATUS(1,14,8,INAME,130,0)          MUL0560C
    INAME=8HCONTACT          MUL0561C
    CALL DSTATUS(1,13,8,INAME,2,0)          MUL0562C
    ICRT1=1          MUL0563C
2203 GO TO 6000          MUL0564C
2300 IF(CONTR(1)=ACROMAX)Z301,Z301,Z302          MUL0565C
2301 INAME=8H          MUL0566C
    CALL DSTATUS(1,11,8,INAME,-254,0)          MUL0567C
    INAME=8HARGET IS          MUL0568C
    CALL DSTATUS(1,12,8,INAME,-126,0)          MUL0569C
    INAME=8H IN RANG          MUL0570C
    CALL DSTATUS(1,13,8,INAME,2,0)          MUL0571C
    INAME=8HE          MUL0572C
    CALL DSTATUS(1,14,8,INAME,130,0)          MUL0573C

```

```

ICRIT5=1
2302 GO TO 2100
2400 IF(I-1)2401,2401,2402
2401 IF(IISHOOT1)2500,2500,2403
2402 IF(IISHOOT2)2500,2500,2404
2403 ISHOOT1=0
      GO TO 2405
2404 ISHOOT2=0
2405 IASROC=1
NOSHOOT=1
      GO TO 7000
2500 IF(I-NRDD)2600,2700,2700
2600 I=I+1
      GO TO 5000
2700 IF(NOSHOOT)3000,3000,2800
2800 IF(IGTIME-TOB)3000,8000,8000
C   RADIATION MODEL 3000
C
3000 IF(NSHOTS)9500,9500,3001
3001 DO 3002 I=1,NRDD
      DO 3002 J=1,NSHOTS
DISTC(I,J)=SQRTF((CLOUDX(I,J)-DDX(I,I))**2+(CLOUDY(I,J)-DDY(I,I))**2)
DISTP(I,J)=SQRTF((POOLX(I,J)-DDX(I,I))**2+(POOLY(I,J)-DDY(I,I))**2)
3002 CONTINUE
      DO 3007 I=1,NRDD
      DO 3006 J=1,NSHOTS
IF(DISTC(I,J)-CLOUDR(I,J))3016,3004
3016 IF((IDEEP-3)3014,3015,3014
3014 TEMP1=.1
      GO TO 3003
3015 TEMP1=.333
3003 TEMPRL(I,J)=1.58*10.0**14*(10.0-LOGF(60.0*(GTIME-TBURST(I,J))))*TEMPMUL6090
11*YIELD/(CLOUDR(I,J)**2*CLOUDZ(I,J))*(2.78/(160.0*(GTIME-TBURST(I,J))))*MUL6100
MUL6110
MUL6120
      GO TO 3017

```

```

3004 TEMP瞿(1,J)=0.0
3017 IF(IDISTP(1,J)-POOLR(J))3005,3005,3018
  3005 TEMP瞿(1,J)=710.0**25*YIELD/(THERMO*POOLR(J)**2)*(50.0/(GTIME-
    TBURST(J))**1.32*EXP(-4.0*DISTR(1,J)**2)
    GO TO 3006
  3018 TEMP瞿(1,J)=0.0
  3006 CONTINUE
  3007 CONTINUE
    DO 3009 I=IONE,NRDD
      SUM=0.0
      DO 3008 J=1,NSHOTS
        SUM=SUM+TEMP瞿(I,J)
  3009 RADRATE(I)=SUM
      DO 3011 I=IONE,NRDD
        SUM=0.0
        DO 3010 J=1,NSHOTS
          SUM=SUM+TEMP瞿(I,J)
  3011 RADRATE(I)=RADRATE(I)+SUM
      DO 3012 I=1,NRDD
        RADDOSE(I)=RADDOSE(I)+.5*RADRATE(I)
  3012 RADDOSE(I)=RADDOSE(I)+.5*RADRATE(I)
  3013 GO TO 9500
C
C SUBMARINE LOGIC MODEL 4000
C
  4000 IF(IISUB)4001,4001,4400
  4001 IF(IISTRAT-1)4100,4200,4300
  4100 IF(ILOGIC-1)4101,4102,4105
  4101 CALL UNIFORM(400.0,300.0,SUBD)
    CALL UNIFORM(15.0,3.0,SUBS)
    CALL UNIFORM(0.0,20.0,SUBC)
    IF(SUBC)4106,4107,4107
  4106 SUBC=SUBC+360.0
  4107 ILOGIC=1
    TLOGIC=GTIME
    GO TO 900
  4102 IF(GTIME-TLOGIC-10.0)4104,4103,4103

```

```

4103 CALL UNIFORM(7.0,4.0,SUBS)
    CALL UNIFORM(0.0,10.0,SUBC)
    IF(SUBC)4108,4109,4109
4108 SUBC=SUBC+360.0
4109 ILOGIC=2
4104 GO TO 900
4105 CALL UNIFORM(0.0,10.0,SUBC)
    IF(SUBC)4110,4111,4111
4110 SUBC=SUBC+360.0
4111 GO TO 900
4200 IF(SSB(1)-90.0)4201,4215
4215 IF(SSB(1)-270.0)4216,4216,4201
4216 IF(SSB(2)-90.0)4201,4217,4217
4217 IF(SSB(2)-270.0)4214,4214,4201
4201 IF(SSB(1)-180.0)4203,4203,4202
4202 TEMP1=360.0-SSB(1)
    GO TO 4204
4203 TEMP1=SSB(1)+180.0
4204 IF(SSB(2)-180.0)4206,4206,4205
4205 TEMP2=360.0-SSB(2)
    GO TO 4207
4206 TEMP2=SSB(2)
4207 ANGLE=ABSF(TEMP1-TEMP2)
    IF(TEMP1-180.0)4208,4208,4210
4208 IF(TEMP2-180.0)4209,4209,4211
4209 CALL UNIFCRM(210.0,10.0,SUBC)
    GO TO 4213
4210 IF(TEMP2-180.0)4212,4211,4211
4211 CALL UNIFORM(150.0,10.0,SUBC)
    GO TO 4213
4212 CENTERB=(TEMP1+TEMP2)/2.0
    SUBC=CENTERB
4213 CALL UNIFORM(8.0,3.0,SUBS)
    CALL UNIFORM(300.0,100.0,SUBD)
4214 GO TO 900
4300 W1=SQRTF((SUBX-DDX(1))**2+(SUBY-DOY(1))**2)

```

```

W2=SQRTF((SUBX-DDX(2))**2+(SUBY-DDY(2))**2)
IF(W1-W2)4301,4301,4302
 4301 RANGE=W1
    J=1
    GO TO 4303
 4302 RANGE=W2
    J=2
 4303 IF(ABSFISSB(J)-B)-2.0)4304,4304,4305
 4304 NCONBER=NCONBER+1
    GO TO 4306
 4305 NCONBER=0
 4306 B=SSB(J)
 4308 IF(ILOGIC-1)4311,4315,4318
 4311 CALL UNIFORM(0.0,1.0,TURN)
    IF(TURN)4312,4312,4313
 4312 ITURN=1
    CALL UNIFORM(135.0,30.0,SUBC)
    GO TO 4314
 4313 ITURN=0
    CALL UNIFORM(225.0,30.0,SUBC)
 4314 CALL UNIFORM(400.0,300.0,SUBD)
    IF(SUBD-SUBDMAX)4344,4344,4343
 4343 SUBD=SUBDMAX
 4344 CALL UNIFORM(15.0,3.0,SUBS)
    IF(SUBS-SUBSMAX)4342,4336,4336
 4336 SUBS=SUBSMAX
 4342 ILOGIC=1
    TLOGIC=GTIME
    GO TO 900
 4315 IF(GTIME-TLOGIC-10.0)4317,4316,4316
 4316 ILOGIC=2
    CALL UNIFORM(7.0,4.0,SUBS)
    IF(SUBS-SUBSMAX)4317,4337,4337
 4337 SUBS=SUBSMAX
 4317 GO TO 900
 4318 IF (ILOGIC-314319,4325,4331

```

```

4319 IF(INCONBER-8)4320,4321,4321
4320 GO TO 900
4321 IF(UTURN)4322,4322,4323
4322 CALL UNIFORM(270.0,30.0,SUBC)
GO TO 4324
4323 CALL UNIFORM(090.0,30.0,SUBC)
CALL UNIFORM(400.0,300.0,SUBD)
IF(SUBD-SUBDMAX)4346,4346,4345
4345 SUBD=SUBDMAX
4346 CALL UNIFORM(7.0,3.0,SUBS)
IF(SUBS-SUBSMAX)4339,4338,4338
4338 SUBS=SUBSMAX
4339 ILOGIC=3
GO TO 900
4325 IF(INCONBER-8)4326,4327,4327
4326 GO TO 900
4327 IF(RANGE=10000.0)4328,4328,4326
4328 IFIRE=1
4330 CALL UNIFORM(180.0,60.0,SUBC)
CALL UNIFORM(400.0,300.0,SUBD)
IF(SUBD-SUBDMAX)4348,4348,4347
4347 SUBD=SUBDMAX
4348 CALL UNIFORM(7.0,4.0,SUBS)
IF(SUBS-SUBSMAX)4341,4340,4340
4340 SUBS=SUBSMAX
4341 ILOGIC=4
VEL=DDS(J)+18.0
TINTER=.03*RANGE/VEL+GTIME
GO TO 900
4331 IF(TINTER-GTIME)4333,4333,4332
4332 GO TO 900
4333 CALL UNIFORM(0.5,0.5,PHIT)
IF(PHIT-.3)4334,4334,4335
4334 IF((J-1)4335,4349,4350
4349 NRDD=NRDO-1
DOX(2)=0.0

```

```

DDY(2)=-0.0          MUL07570
DDS(2)=0.0           MUL07580
GO TO 4351          MUL07590
4350 IONE=IONE+1    MUL07600
DDX(1)=-0.0          MUL07610
DDY(1)=-0.0          MUL07620
DDS(1)=0.0           MUL07630
IF(NRDD-IONE)4352,4353,4353
4351 ICRIT14=1      MUL07640
4352 ICRIT14=1      MUL07650
IEND=1              MUL07660
GO TO 800            MUL07670
4353 INAME=8H        DD
CALL DSTATUS(1,12,8,INAME,-126,0)
INAME=8H
CALL DSTATUS(1,14,8,INAME,130,0)
CALL DSTATUS(1,11,8,INAME,-254,0)
INAME=BHSUNK
CALL DSTATUS(1,13,8,INAME,2,0)
ICRIT7=1
4335 ILOGIC=3        MUL07750
GO TO 900            MUL07760
4400 IF(ILOGIC)4425,4425,4428
4425 SJ2(4401)       MUL07770
4426 SJ3(4427)       MUL07780
GO TO 900            MUL07790
4427 ILOGIC=1        MUL07800
ICRIT3=1
VEL=DDS(J)+18.0      MUL07810
TINTER=.03*RANGE/VEL+GTIME
GO TO 900            MUL07820
4401 DO 4402 I=1,NRDD
4402 SB(1)=8.4*DDS(1)
DO 4414 I=1,2
DX=DDX(1)-SUBX
DY=DDY(1)-SUBY
IF(DX) 4405,4403,4404

```

```

4403 IF(DY) 4410,4410,4409          MUL07930
4404 IF(DY) 4408,4406,4407          MUL07940
4405 IF(DY) 4412,4411,4413          MUL07950
4406 DDB(1)=090.                      MUL07960
        GO TO 4414                      MUL07970
4407 DDB(1)=90.-ATANF( DY/DX)*57.295   MUL07980
        GO TO 4414                      MUL07990
4408 DDB(1)=90.+ATANF(-DY/DX)*57.295    MUL08000
        GO TO 4414                      MUL08010
4409 DDB(1)=000.0                      MUL08020
        GO TO 4414                      MUL08030
4410 DDB(1)=180.0                      MUL08040
        GO TO 4414                      MUL08050
4411 DDB(1)=270.0                      MUL08060
        GO TO 4414                      MUL08070
4412 DDB(1)=270.0-ATANF(DY/DX)*57.295   MUL08080
        GO TO 4414                      MUL08090
4413 DDB(1)=270.0+ATANF(-DY/DX)*57.295   MUL08100
4414 CONTINUE                         MUL08110
        WRITE OUTPUT TAPE 9,4424,DDB(1)*SB(1)*DDB(2)*SB(2)
4424 FORMAT(6X,10HDD1 BEARS ,F4.0,12H SCREW BEAT ,F4.0/6X,10HDD2 BEARS
1,F4.0,12H SCREW BEAT ,F4.0/1)      MUL08120
        WRITE OUTPUT TAPE 9,4415          MUL08130
4415 FORMAT(6X,13HCOURSE ORDERS/1
READ INPUT TAPE 9.4416,SUBC         MUL08140
        WRITE OUTPUT TAPE 9,4417,SBBC     MUL08150
4417 FORMAT(6X,15HNEW SUB COURSE ,F4.0/1) MUL08160
        WRITE OUTPUT TAPE 9,4418          MUL08170
4418 FORMAT(6X,12HSPEED ORDERS/)      MUL08180
        READ INPUT TAPE 9.4419,SUBS     MUL08190
4419 FORMAT(F3.0)                     MUL08200
        WRITE OUTPUT TAPE 9,4420,SUBS     MUL08210
4420 FORMAT(6X,14HNEW SUB SPEED ,F3.0/1) MUL08220
        WRITE OUTPUT TAPE 9,4421          MUL08230
4421 FORMAT(6X,12HDEPTH ORDERS/)      MUL08240

```

```

READ INPUT TAPE 9,4422, SUBD
4422 FORMAT(F5.0)
WRITE OUTPUT TAPE 9,4423, SUBD
4423 FORMAT(6X,14HNEW SUB DEPTH ,F5.0/)
GO TO 4426
4428 IF(TINTER-GTIME)4430,4430,4429
4429 GO TO 4425
4430 CALL UNIFORM(.5,.5,PHIT)
IF(PHIT-.3)4431,4431,4432
4431 W1=SQRTF((SUBX-DDX(1))*2+(SUBY-DDY(1))*2)
W2=SQRTF((SUBX-DDX(2))*2+(SUBY-DDY(2))*2)
IF(W1-W2)4435,4435,4436
4435 IONE=IONE+1
GO TO 4437
4436 NRDD=NRDD-1
4437 IF(NRDD-IONE)4438,4439,4439
4438 ICRT14=1
IEND=1
GO TO 800
4439 CONTINUE
4432 ILOGIC=0
GO TO 4425
C SONAR CONTACT MODEL 5000
C
5000 DETRM=33.3*SQRTF(-SUBD/(GRAD*.05+.018))
SIGMA=.3*DETRM
CALL NORMAL(DETRM,SIGMA,DETR)
IF(ISS-2)5029,5030,5028
5028 IF(ISS-4)5031,5032,5033
5029 IF(DDS(1)-22.0)5034,5034,5035
5030 IF(DDS(1)-20.0)5034,5034,5035
5031 IF(DDS(1)-18.0)5034,5034,5035
5032 IF(DDS(1)-17.0)5034,5034,5035
5033 IF(DDS(1)-16.0)5034,5034,5035
5034 DETR=.7*DETR
      
```

```

5035 RANGE=SQRTF((DDX(1)-SUBX)**2+(DDY(1)-SUBY)**2)
      IF(RANGE-DETR)5002,5001,5001
5001 IDOPLER(1)=0
      MARKS(1)=0
      NMARKS(1)=0
      ICLASS(1)=0
      GO TO 2500
5002 SIGMA=.04*RANGE
      CALL NORMAL(RANGE,SIGMA,CONTR(1))
      DY=SUBY-DDY(1)
      DX=SUBX-DDX(1)
      IF(DX) 5005,5003,5004
      5003 IF(DY) 5010,5010,5009
      5004 IF(DY) 5008,5006,5007
      5005 IF(DY) 5012,5011,5013
      5006 SSB(1)=090.
      GO TO 5014
      5007 SSB(1)=90.-ATANF( DY/DX)*57.295
      GO TO 5014
5008 SSB(1)=90.+ATANF(-DY/DX)*57.295
      GO TO 5014
5009 SSB(1)=000.0
      GO TO 5014
5010 SSB(1)=180.0
5011 SSB(1)=270.0
      GO TO 5014
5012 SSB(1)=270.0-ATANF(DY/DX)*57.295
      GO TO 5014
5013 SSB(1)=270.0+ATANF(-DY/DX)*57.295
5014 D*SSB(1)-SUBC
      DA*ARSF(0)
      IF(IDA-.45*.015015,5015,5016
5015 IDOPLER(1)-2
      GO TO 5019
5016 IF(IDA-135.015017,5018,5018
      MUL08650
      MUL08660
      MUL08670
      MUL08680
      MUL08690
      MUL08700
      MUL08710
      MUL08720
      MUL08730
      MUL08740
      MUL08750
      MUL08760
      MUL08770
      MUL08780
      MUL08790
      MUL08800
      MUL08810
      MUL08820
      MUL08830
      MUL08840
      MUL08850
      MUL08860
      MUL08870
      MUL08880
      MUL08890
      MUL088900
      MUL088910
      MUL088920
      MUL088930
      MUL088940
      MUL088950
      MUL088960
      MUL088970
      MUL088980
      MUL088990
      MUL089000

```

```

5017 IDOPLER(1)=0          MUL09010
      GO TO 5019          MUL09020
5018 IDOPLER(1)=1          MUL09030
5019 SIGMA=4.5            MUL09040
      CALL NORMAL(ISSB(1)),SIGMA,CONTB(1)  MUL09050
      IF (CONTB(1)) 5020,5021,5021
5020 CONTB(1)=CONTB(1)+360.0  MUL09060
      GO TO 5023          MUL09070
5021 IF (CONTB(1)-360.0) 5023,5022,5022  MUL09070
5022 CONTB(1)=CONTB(1)-360.0  MUL09100
5023 NMARKS(1)=NMARKS(1)+1  MUL09110
      IF (NMARKS(1)-2) 5027,5025,5024  MUL09120
5024 IF (NMARKS(1)-10) 5025,5026,5026  MUL09130
5025 ICLASS(1)=1          MUL09140
      GO TO 5027          MUL09150
5026 ICLASS(1)=2          MUL09160
5027 ICONTB(1)=CONTB(1)  MUL09170
      GO TO 220           MUL09180
      C CONTACT TRACKING MODEL 6000  MUL09190
      C
6000 IF (MARKS(1)-5) 6001,6002,6002  MUL09200
6001 MARKS(1)=MARKS(1)+1  MUL09210
6002 CONTX(1)=DDX(1)+CONTR(1)*SINF((CONTB(1)/57.296)  MUL09220
      CONTRY(1)=DDY(1)+CONTR(1)*COSF((CONTB(1)/57.296)  MUL09230
      IF (MARKS(1)-3) 6003,6004,6004  MUL09240
6003 IF (MARKS(1)-1) 6005,6006,6007  MUL09250
6004 X(1,1)=CONTX(1)  MUL09260
      Y(1,1)=CONTY(1)  MUL09270
      GO TO 6005          MUL09280
6005 X(1,2)=X(1,1)  MUL09290
      Y(1,2)=Y(1,1)  MUL09300
      X(1,1)=CONTX(1)  MUL09310
      Y(1,1)=CONTY(1)  MUL09320
6006 GO TO 2500          MUL09330
6004 NPTSP1=MARKS(1)+1  MUL09340
                                MUL09350
                                MUL09360

```

```

NPTSP2=MARKS(II)+2
N=MARKS(1)
DO 6023 J=2,N
  X(I,NPTSP2-J)=X(I,NPTSP1-J)
  Y(I,NPTSP2-J)=Y(I,NPTSP1-J)
6023 X(I,1)=CONTX(I)
  Y(I,1)=CONTY(I)
  DX=X(I,1)-X(I,N)
  DY=Y(I,1)-Y(I,N)
  DR=SQR(DX**2+DY**2)
  BNPTS=MARKS(I)
  CONTS(I)=.06*DR/(BNPTS-1.0)
  IF(NMARKS(I)-8)6027,6028,6026
6026 IF(NMARKS(I)-10)6029,6030,6031
6027 ISOL(I)=1
  GO TO 6032
6028 ISOL(I)=2
  GO TC 6032
6029 ISOL(I)=3
  GO TO 6032
6030 ISOL(I)=4
  GO TO 6032
6031 ISOL(I)=5
  GO TO 6032
6032 CONTINUE
  IF(DX)6008,6009,6010
6C08 IF(DY)6014,6015,6016
6009 IF(DY)6017,6018,6018
6010 IF(DY)6011,6012,6013
6011 THETA=ATANF(-DY/DX)
  CONT(I)=90.0+THETA*57.296
  GO TO 6012
6012 CONT(I)=090.0
  GO TO 6019
6013 THETA=ATANF(DY/DX)
  CONT(I)=C(J,0..THETA*57.296)
  GO TO 6019

```

```

6014 THETA=ATAN((Y/X))      J.U-IHETA*57.296
    CONTC(1)=
60 TO 6019
6015 CONTC(1)=270.0
    GO TO 6019
6016 CONTC(1)=ATANF(-DY/DX)
    CONTC(1)=270.0+THETA*57. 96
    GO TO 6019
6017 CONTC(1)=180.0
    GO TO 6019
6018 CONTC(1)=000.0
6019 ICONT(1)=CONT(1)
    IF(MARKS(1)-4)6005,6020,6020
6020 IF(CTRL(1)-AROCMAX)6021,6035,6035
6021 IF(CTRL(1)-AROCMIN)6033,6072,6022
6022 GO TO 2300
6033 INAME=8H TARGET
    CALL DSTATUS(1,11,8,INAME,-254,0)
    INAME=8H TOO CLO
    CALL DSTATUS(1,12,8,INAME,-126,0)
    INAME=6HST TO SH
    CALL DSTATUS(1,13,8,INAME,2,0)
    INAME=8HROOT AT
    CALL DSTATUS(1,14,8,INAME,130,0)
    ICPI18=1
    GO TO 2500
6035 INAME=8H TARGE
    INAME=8H OUT OF
    CALL DSTATUS(1,12,8,INAME,-126,0)
    INAME=8H ASROC R
    CALL DSTATUS(1,13,8,INAME,2,0)
    INAME=8HANGE
    CALL DSTATUS(1,14,8,INAME,130,0)
    ICRI9=1
    GO TO 2500
MUL09730
MUL09740
MUL09750
MUL09760
MUL09770
MUL09780
MUL09790
MUL097A0
MUL097B0
MUL097C0
MUL097D0
MUL097E0
MUL097F0
MUL09800
MUL09810
MUL09820
MUL09830
MUL09840
MUL09850
MUL09860
MUL09870
MUL09880
MUL09890
MUL098A0
MUL098B0
MUL098C0
MUL098D0
MUL098E0
MUL098F0
MUL09900
MUL10010
MUL10020
MUL10030
MUL10040
MUL10050
MUL10060
MUL10070
MUL10080

```

```

C WEAPON FIRING MODEL 7000
C
7000 CALL UNIFORM(•5,•5,REL)
    IF(REL•9)7009,7019,7008
7008 INAME•8H ASROC
    CALL DSTATUS(1,12,8,INAME,-126,0)
    INAME•8H
    CALL DSTATUS(1,11,8,INAME,-254,0)
    CALL DSTATUS(1,14,6,INAME,130,0)
    INAME•8H MISFIRE
    CALL DSTATUS(1,13,8,INAME,2,0)
    ICRT4=1
    WRITE OUTPUT TAPE 5,7010
7010 FORMAT(13HASROC MISFIRE)
    GO TO 2500
7009 TOF•064•R•20.0
    TOS=DOB/SR
    DTIME=TOF+TOS
    IF(DTIME-45.0)17001,7002
    17002 IF(DTIME-75.0)17003,7003
    7004 IF(DTIME-105.0)17005,7005
    7001 DTIME=.5
    GO TO 7007
    7003 DTIME=1.0
    GO TO 7007
    7005 DTIME=1.5
    GO TO 7007
    7006 DTIME=2.0
    7007 XTEMP•CONTX(1)+16.667•CONTS(1)*SINF(ONTC(1))
        YTEMP•CONTY(1)+16.667•CONTS(1)*COSF(ONTC(1))
        SIGMA=1.2+(3.1/800.0)*CONT(1)
        CALL ERROR(XTEMP,YTEMP,SIGMA,G2X,G2Y)
        TOB=DTIME+DTIME
        INAME•8H
        CALL DSTATUS(1,11,8,INAME,-254,0)
        CALL DSTATUS(1,14,8,INAME,130,0)

```

```

INAME=8H ASROC
CALL DSTATUS(1,12,8,INAME,-126,0)
INAME=8H FIRED
CALL DSTATUS(1,13,8,INAME,2,0)
ICRIT2=1
WRITE OUTPUT TAPE 5,7011,I
7011 FORMAT (2HDD,11,12H FIRED ASROC)
GO TO 2500

C
C EVALUATION MODEL 8000
C
8000 NSHOTS=NSHOTS+1
NOSHOOT=0
TBURST(NSHOTS)=TOB
POOLX(NSHOTS)=GZX
POOLY(NSHOTS)=GZY
CLOUDX(NSHOTS)=GZX
CLOUDY(NSHOTS)=GZY
CLOUDZ(NSHOTS)=2000.0*(YIELD/10.0)**.167
TEMP1=DOB**1.33/(1500.0*YIELD**.333)
IF(TEMP1-1.0)>0002,8001,8001
8001 POOLR(NSHOTS)=1580.0*(YIELD/TEMP1)**.25
GJ TO 8003
8002 POOLR(NSHOTS)=1580.0*YIELD**.25
8003 CONTINUE
TEMP1=SQRTF((17100000.0*YIELD*(STRESS*HULL)))
RLETHAL=(CRUSH-300.0)*TEMP1/(CRUSH-SUBD)
TEMP2=SQRTF((GZX-SUBX)**2+(GZY-SUBY)**2+((SUBD-DOB)/3.0)**2)
IF(TEMP2-RLETHAL)>0004,8004,8006
8004 INAME=8H SUB
CALL DSTATUS(1,12,8,INAME,-126,0)
INAME=8H
CALL DSTATUS(1,11,8,INAME,-254,0)
CALL DSTATUS(1,14,8,INAME,130,0)
INAME=8HSUNK
CALL DSTATUS(1,13,8,INAME,2,0)

```

```

ICRIT6=1          MUL10310
IEND=1          MUL10520
IF(IASROC-1)8005,8005,8010
ICRIT15=1          MUL10530
GO TO 8011          MUL10540
MUL10550
MUL10560
MUL10570
MUL10580
MUL10590
MUL10600
MUL10610
MUL10620
MUL10630
MUL10640
MUL10650
MUL10660
MUL10670
MUL10680
MUL10690
MUL10700
MUL10710
MUL10720
MUL10730
MUL10740
MUL10750
MUL10760
MUL10770
MUL10780
MUL10790
MUL10800
MUL10810
MUL10820
MUL10830
MUL10840
MUL10850
MUL10860
MUL10870
MUL10880
MUL10890
MUL10900
MUL10910
MUL10920
MUL10930
MUL10940
MUL10950
MUL10960
MUL10970
MUL10980
MUL10990
MUL11000
MUL11010
MUL11020
MUL11030
MUL11040
MUL11050
MUL11060
MUL11070
MUL11080
MUL11090
MUL11100
MUL11110
MUL11120
MUL11130
MUL11140
MUL11150
MUL11160

8005 ICRIT16=1
8010 ICRIT16=1
8011 GO TO 800
8006 IF(TEMP2-2.0*RLETHAL)8008,8008,8007
8007 GO TO 3000
8008 DAMAGE=2.0-TEMP2/RLETHAL
DAMAGE T=DAMAGE T+DAMAGE
IF(DAMAGE T-.75)8009,8C09,8004
8009 SUBDMAX=(1.0-DAMAGE)*CRUSH*SAFETY
SUBSMAX=(1.0-DAMAGE)*SUBSMAX
GO TO 3000

C CRITIQUE 11 9000
C
9000 IF(ICRIT11)9001,9001,9011
9001 IF(ICRIT12)9002,9002,9012
9002 IF(ICRIT13)9003,9003,9013
9003 IF(ICRIT14)9004,9004,9014
9004 IF(ICRIT15)9005,9005,9015
9005 IF(ICRIT16)9006,9006,9016
9006 IF(ICRIT17)9007,9007,9017
9007 GO TO 700
9011 WRITE OUTPUT TAPE 5,9021
GO TO 9018
9012 WRITE OUTPUT TAPE 5,9022
GO TO 9018
9013 WRITE OUTPUT TAPE 5,9023
GO TO 9018
9014 WRITE OUTPUT TAPE 5,9024
GO TO 9018
9015 WRITE OUTPUT TAPE 5,9025
GO TO 9018

```

```

9016 WRITE OUTPUT TAPE 5,9026
    GO TO 9018
9017 WRITE OUTPUT TAPE 5,9027
    GO TO 700
9021 FORMAT(27HGAME IS A DRAW, SUB ESCAPED)
9022 FORMAT(34HSUB WINS. DD1 SUNK AND SUB ESCAPED)
9023 FORMAT(34HSUB WINS. DD2 SUNK AND SUB ESCAPED)
9024 FORMAT(30HSUB WINS. BOTH DESTROYERS SUNK)
9025 FORMAT(24HDD WINS. SUB SUNK BY DD1)
9026 FORMAT(24HDD WINS. SUB SUNK BY DD2)
9027 FORMAT(30HSUB WINS BY AVOIDING DETECTION)
MUL11170
MUL11180
MUL11190
MUL11200
MUL11210
MUL11220
MUL11230
MUL11240
MUL11250
MUL11260
MUL11270
MUL11280
MUL11290
MUL11300
MUL11310
MUL11320
MUL11330
MUL11340
MUL11350
MUL11360
MUL11370
MUL11380
MUL11390
MUL11400
MUL11410
MUL11420
MUL11430
MUL11440
MUL11450
MUL11460
MUL11470
MUL11480
MUL11490
MUL11500
MUL11510
MUL11520

C CRITIQUE 1 9500
C
9500 IF(MARKS(1)) 9502,9502,9504
9502 IF(MARKS(2)) 9511,9511,9503
9503 IF(MARKS(2)-3)9512,9513,9513
9504 IF(MARKS(2)) 9505,9505,9506
9505 IF(MARKS(1)-3) 9515,9514,9514
9506 IF(MARKS(1)-3)9508,9507,9507
9507 IF(MARKS(2)-3)9517,9516,9516
9508 IF(MARKS(2)-3)9519,9518,9518
9511 WRITE OUTPUT TAPE 5,9521,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
1((1),I=1,NRDD),(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD))
GO TO 9599
9512 WRITE OUTPUT TAPE 5,9522,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
1((1),I=1,NRDD),ICLASS(2),CONTR(2),ICONTB(2),IDOPLER(2),
2(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD))
GO TO 9599
9513 WRITE OUTPUT TAPE 5,9523,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
1((1),I=1,NRDD),ICLASS(2),CONTR(2),ICONTB(2),IDOPLER(2),
2ICONTC(2),ICONTS(2),ISOL(2),
3(RADRATE(1),I=1,NRDD),(RADDOSE(1),I=1,NRDD))
GO TO 9599
9514 WRITE OUTPUT TAPE 5,9524,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
1((1),I=1,NRDD),ICLASS(1),CONTR(1),ICONTB(1),IDOPLER(1)),

```

```

      2ICONTC(1),CONTS(1),ISOL(1),
      3(RADRATE(1),I=1,NRDD),(RADDSE(1),I=1,NRDD)
      GO TO 9599

9515 WRITE OUTPUT TAPE 5,9525,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
      1(1),I=1,NRDD),(ICLASS(1),CONT(1),ICONTB(1),IDOPLER(1),
      2(RADRATE(1),I=1,NRDD),(RADDSE(1),I=1,NRDD)
      GO TO 9599

9516 WRITE OUTPUT TAPE 5,9526,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
      1(1),I=1,NRDD),(ICLASS(1),I=1,NRDD),(CONT(1),I=1,NRDD),(ICONTB(1),
      2,I=1,NRDD),(IDOPLER(1),I=1,NRDD),
      3(ICONIC(1),I=1,NRDD),(CONTS(1),I=1,NRDD),(ISOL(1),I=1,NRDD),
      4(RADRATE(1),I=1,NRDD),(RADDSE(1),I=1,NRDD)
      GO TO 9599

9517 WRITE OUTPUT TAPE 5,9527,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
      1(1),I=1,NRDD),(ICLASS(1),I=1,NRDD),(CONT(1),I=1,NRDD),(ICONTB(1),
      2,I=1,NRDD),(IDOPLER(1),I=1,NRDD),
      3ICONTC(1),CONTS(1),ISOL(1),
      4(RADRATE(1),I=1,NRDD),(RADDSE(1),I=1,NRDD)
      GO TO 9599

9518 WRITE OUTPUT TAPE 5,9528,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
      1(1),I=1,NRDD),(ICLASS(1),I=1,NRDD),(CONT(1),I=1,NRDD),(ICONTB(1),
      2,I=1,NRDD),(IDOPLER(1),I=1,NRDD),
      3ICONIC(2),CONTS(2),ISOL(2),
      4(RADRATE(1),I=1,NRDD),(RADDSE(1),I=1,NRDD)
      GO TO 9599

9519 WRITE OUTPUT TAPE 5,9529,ESR,IWINDD,GTIME,WINDV,YIELD,ISS,(DDSMAX
      1(1),I=1,NRDD),(ICLASS(1),I=1,NRDD),(CONT(1),I=1,NRDD),(ICONTB(1),
      2,I=1,NRDD),(IDOPLER(1),I=1,NRDD),
      3(RADRATE(1),I=1,NRDD),(RADDSE(1),I=1,NRDD)
      GO TO 9599

9521 FORMAT 15X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X,
      113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X,
      212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL1185C
      35X,15HMAX SPEED AVAIL,6X,F3.0,3X,F3.0/
      45X,13HCONTACT CLASS/5X,1HSOAR RANGE/5X,13HSOAR BEARING/5X,
      57HDOPPLER/5X,13HTARGET COURSE/5X,12HTARGET SPEED/5X,15HFIRING SOLUMUL11880

```

6ITION/5X,14HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE, . MUL11890  
75X,F5.0,6X,F5.0) MUL11900

9522 FORMAT( 5X,15SHEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL11910  
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL11920  
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL11930  
35X,15HMAX SPEED AVAIL,6X,F3.0,0/ MUL11940  
45X,13HCCONTACT CLASS,20X,11/5X,1HSOUNAR RANGE,18X,F6.0/5X,13HSOUNAR MUL11950  
5BEARING,18X,13/5X,7HDOPPLER,26X,11/5X,13HTARGET COURSE/5X, MUL11960  
612HTARGET SPEED/5X,15HFIRING SOLUTION/5X, MUL11970  
714HRADIATION RATE,6X,F4.0/5X,14HRADIATION DOSE,5X,F5.0,0, MUL11980  
86X,F5.0) MUL11990

9523 FORMAT( 5X,15SHEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL12000  
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL12010  
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL12020  
35X,15HMAX SPEED AVAIL,6X,F3.0,0/ MUL12030  
45X,13HCCONTACT CLASS,20X,11/5X,1HSOUNAR RANGE,18X,F6.0/5X,13HSOUNAR MUL12040  
5BEARING,18X,13/5X,7HDOPPLER,26X,11/5X,13HTARGET COURSE,18X,13/5X, MUL12050  
612HTARGET SPEED,20X,F3.0/5X,15HFIRING SOLUTION,18X,11/5X, MUL12060  
714HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0,0, MUL12070  
86X,F5.0) MUL12080

9524 FORMAT( 5X,15SHEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL12090  
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL12100  
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL12110  
35X,15HMAX SPEED AVAIL,6X,F3.0,0/ MUL12120  
45X,13HCCONTACT CLASS, 9X,11/5X,1HSOUNAR RANGE, 7X,F6.0/5X,13HSOUNAR MUL12130  
5BEARING, 7X,13/5X,7HDOPPLER,15X,11/5X,13HTARGET COURSE, 7X,13/5X, MUL12140  
612HTARGET SPEED, 9X,F3.0/5X,15HFIRING SOLUTION, 7X,11/5X, MUL12150  
714HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0,0, MUL12160  
86X,F5.0) MUL12170

9525 FORMAT( 5X,15SHEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL12180  
113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL12190  
212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL12200  
35X,15HMAX SPEED AVAIL,6X,F3.0,0/ MUL12210  
45X,13HCCONTACT CLASS, 9X,11/5X,1HSOUNAR RANGE, 7X,F6.0/5X,13HSOUNAR MUL12220  
5BEARING, 7X,13/5X,7HDOPPLER,15X,11/5X,13HTARGET COURSE/5X, MUL12230  
612HTARGET SPEED/5X,15HFIRING SOLUTION/5X, MUL12240

714HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0, MUL12255  
 86X,F5.0) MUL12260

**9526 FORMAT(** 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL12270  
 113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL12280  
 212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL12299  
 35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F2.0/ MUL12300

45X,13HCONTACT CLASS,9X,11,10X,11/5X,11HSOINAR RANGE,7X,F6.0,5X,F6.0 MUL12310  
 5/5X,13HSOINAR BEARING,7X,13,8X,13/5X,7HDOPPLER,15X,11,10X,11/ MUL12320  
 65X,13HTARGET COURSE,7X,13,6X,13/5X,12HTARGET SPEED,9X,F3.0,8X,F3.0 MUL12330  
 7/5X,15HFIRING SOLUTION,7X,11,10X,11/5X,  
 814HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0, MUL12340  
 96X,F5.0) MUL12350

**9527 FORMAT(** 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL12360  
 113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL12370  
 212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL12380  
 35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F2.0/ MUL12403

45X,13HCONTACT CLASS,9X,11,10X,11/5X,11HSOINAR RANGE,7X,F6.0,5X,F6.0 MUL12410  
 5/5X,13HSOINAR BEARING,7X,13,8X,13/5X,7HDOPPLER,15X,11,10X,11/ MUL12420  
 65X,13HTARGET COURSE,7X,13,5X,12HTARGET SPEED,9X,F3.0 MUL12420  
 7/5X,15HFIRING SOLUTION,7X,11,5X,  
 814HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0, MUL12440  
 96X,F5.0) MUL12450

**9528 FORMAT(** 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL12460  
 113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL12470  
 212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL12490  
 35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F2.0/ MUL12500

45X,13HCONTACT CLASS,9X,11,10X,11/5X,11HSOINAR RANGE,7X,F6.0,5X,F6.0 MUL12510  
 5/5X,13HSOINAR BEARING,7X,13,8X,13/5X,7HDOPPLER,15X,11,10X,11/ MUL12520  
 65X,13HTARGET COURSE,18X,13,5X,12HTARGET SPEED,20X,F3.0 MUL12530  
 7/5X,15HFIRING SOLUTION,18X,11,5X,  
 814HRADIATION RATE,6X,F4.0,7X,F4.0/5X,14HRADIATION DOSE,5X,F5.0, MUL12540  
 96X,F5.0) MUL12550

**9529 FORMAT(** 5X,15HEFF SONAR RANGE,3X,F6.0,8X,14HWIND DIRECTION,6X, MUL12570  
 113/5X,9HGAME TIME,9X,F7.1,7X,13HWIND VELOCITY,8X,F3.0/5X, MUL12580  
 212HWARHEAD SIZE,9X,F4.1,7X,9HSEA STATE,13X,11/25X,3HDD1,8X,3HDD2/MUL12590  
 35X,15HMAX SPEED AVAIL,6X,F3.0,8X,F2.0/ MUL12600

```

45X,13HCONTACT CLASS,9X,I1,10X,I1/5X,1IHSONAR RANGE,7X,F6,0,5X,F6,0MUL12510
5/5X,13HSONAR BEARING,7X,I3,8X,I3/5X,7HDOPPLER,15X,I1,10X,I1/MUL12620
65X,13HTARGET COURSE/5X,12HTARGET SPEED/5X,15HFIRING SOLUTION/SX,MUL12630
714HRADIATION RATE,6X,F4,0,7X,F4,0/5X,14HRADIATION DOSE,5X,F5,0,MUL12640
86X,F5,0,MUL12650
9599 CONTINUUM
9600 ISUBC=SUBC
DO 9601 I=1,NRDD
 9601 IDDC(1)=DDC(1)
9602 WRITE OUTPUT TAPE 5,9620
9602 WRITE OUTPUT TAPE 5,9621,(DDX(1),I=1,NRDD),SUBX,(DOY(1),I=1,NRDD),MUL12660
9603 ISUBY,(IDDC(1),I=1,NRDD),ISUBC,(DDS(1),I=1,NRDD),SUBS,SUBD
9603 IF (NSHOTS)19604,9604,9605
9604 GO TO 9628
9605 WRITE OUTPUT TAPE 5,9625
DO 9606 I=1,NSHOTS
9606 WRITE OUTPUT TAPE 5,9626 ,I,POOLX(1),POOLY(1),POOLR(1)
9607 WRITE OUTPUT TAPE 5,9627
DO 9608 I=1,NSHOTS
9608 WRITE OUTPUT TAPE 5,9628,I,CLOUDX(1),CLOUDY(1),CLOUDR(1)
9620 FORMAT (1BX,3HDD1,10X,3HDD2,10X,3HSUB)
9621 FORMAT (7X,1HX,7X,F8,1,5X,F8,1,5X,F8,1/7X,1HY,7X,1HY,7X,F8,1,5X,F8,1,5X,F
18,1/5X,CHCOURSE,8X,I3,10X,I3,10X,I3/5X,5HSPEED,9X,F4,1,9X,
2F4,1,9X,F4,1/5X,9HSUB DEPTH,F8,1)
9625 FORMAT (10X,4HPOOL,6X,1HX,9X,1HY,9X,6HADIUS)
9626 FORMAT (10X,I1,(9X,3F8,1))
9627 FORMAT (10X,5HCLQD,5X,1HX,9X,1HY,9X,6HADIUS)
9628 FORMAT (10X,I1,(9X,3F8,1))
9629 WRITE OUTPUT TAPE 5,9630,DAMAGET
9630 FORMAT (19HTOTAL DAMAGE TO SUB,2X,F4,1/1)
IF (ICRIT1)9631,9631,9641
9631 IF (ICRIT2)9632,9632,9642
9632 IF (ICRIT3)9633,9633,9643
9633 IF (ICRIT4)9634,9634,9644
9634 IF (ICRIT5)9635,9635,9645
9635 IF (ICRIT6)9636,9636,9646

```

```

9636 IF(IICRIT7)9637,9637,9647
9637 IF(IICRIT8)9638,9638,9648
9638 IF(IICRIT9)9660,9660,9649
9641 WRITE OUTPUT TAPE 5,9651
    GO TO 9660
9642 WRITE OUTPUT TAPE 5,9652
    GO TO 9660
9643 WRITE OUTPUT TAPE 5,9653
    GO TO 9660
9644 WRITE OUTPUT TAPE 5,9654
    GO TO 9660
9645 WRITE OUTPUT TAPE 5,9655
    GO TO 9660
9646 WRITE OUTPUT TAPE 5,9656
    GO TO 9660
9647 WRITE OUTPUT TAPE 5,9657
    GO TO 9660
9648 WRITE OUTPUT TAPE 5,9658
    GO TO 9660
9649 WRITE OUTPUT TAPE 5,9659
9651 FORMAT(13HSONAR CONTACT/)
9652 FORMAT(11HASROC FIRE/)
9653 FORMAT(13HTORPEDO FIRED/)
9654 FORMAT(13HASROC MISFIRE/)
9655 FORMAT(15HTARGET IN RANGE/)
9656 FORMAT(1HSUB SUNK/)
9657 FORMAT(7HDD SUNK/)
9658 FORMAT(26HTARGET TOO CLOSE TO SHOOT AT//)
9659 FORMAT(25HTARGET CII OF ASROC RANGE//)
9660 ICRIT1=0
9661 ICRIT2=0
9662 ICRIT3=0
9663 ICRIT4=0
9664 ICRIT5=0
9665 ICRIT6=0
9666 ICRIT7=0

```

```

9667 ICRIT8=0
9668 ICRIT9=0
9700 DO 9701 I=1,2
      2ZX(I)=DDX(I)
9701 2ZY(I)=DDY(I)
      2ZX(3)=SUBX
      2ZY(3)=SUBY
      IF(KGRAF>9702,9702,9704
9702 LABEL=8HDD1 SUB
      DO 9703 I=1,12
9703 ITITLE=8H
      CALL DRAW(3,2ZX,2ZY,1,2,LABEL,ITITLE,10000.0,10000.0,4,4,2,2,8,8,
      1,LAST)
      GO TO 9706
9704 KGRAF=KGRAF+1
      IF(KGRAF-619706,9706,9705
9705 KGRAF=1
      ITITLE=8H
      LABEL=8H
      CALL DRAW(3,2ZX,2ZY,2,2,LABEL,ITITLE,10000.0,10000.0,4,4,2,2,8,8,
      1,LAST)
9706 CONTINUE
      GO TO 700
850 END
      SUBROUTINE RANVAR
      COMMON IRANDOM,RANDOM
      CON(K1=67108864)
      IRANDOM=IRANDOM*3125
      IF((IRANDOM-6710886312,2,1
      1 ENQ(0),LDA(IRANDOM),DV1(K1),STO(IRANDOM)
      2 Y=IRANDOM
      RANDOM=Y/67108864.0
      RETURN
END
      SUBROUTINE NORMAL(A,B,C)
      COMMON IRANDOM,RANDOM
      MUL13330
      MUL13340
      MUL13350
      MUL13360
      MUL13370
      MUL13380
      MUL13390
      MUL13400
      MUL13410
      MUL13420
      MUL13430
      MUL13440
      MUL13450
      MUL13460
      MUL13470
      MUL13480
      MUL13490
      MUL13500
      MUL13510
      MUL13520
      MUL13530
      MUL13540
      MUL13550
      MUL13560
      MUL13570
      MUL13580
      MUL13590
      MUL13600
      MUL13510
      MUL13620
      MUL13630
      MUL13640
      MUL13650
      MUL13660
      MUL13670
      MUL13680

```

```

SUM=0.0
DO 1 I=1,12
CALL RANVAR
1 SUM=SUM+RANDOM
X=SUM-6.0
C=X*B+A
RETURN
END
SUBROUTINE UNIFORM(A,B,C)
COMMON RANDOM,RANDOM
CALL RANVAR
C=2.0*B*(RANDOM-.5)+A
RETURN
END
SUBROUTINE ERROR(A,B,C,D,E)
CALL UNIFORM(180.0,179.9,THETA)
CALL NORMAL(0.0,C,R)
R=ABSF(R)
D=A+R*SINF(THETA/57.295)
E=B+R*COSF(THETA/57.295)
RETURN
END
SUBROUTINE CIRCLE(A,B,C,D,E,F,K,L)
DIMENSION U(24),V(24),K(24),L(24)
U(1)=.991*C*.255.0/F
U(2)=.924*C*.255.0/F
U(3)=.793*C*.255.0/F
V(1)=.131*C*.255.0/F
V(2)=.383*C*.255.0/F
V(3)=.609*C*.255.0/F
DO 1 I=4,6
U(I)=V(I-1)
1 V(I)=U(I-1)
DO 2 I=7,9
U(I)=-U(I-1),
2 V(I)=V(I-1)
MUL12690
MUL13700
MUL13710
MUL13720
MUL13730
MUL13740
MUL13750
MUL13760
MUL13770
MUL13780
MUL13790
MUL13800
MUL13810
MUL13820
MUL13830
MUL13840
MUL13850
MUL13860
MUL13870
MUL13880
MUL13890
MUL13900
MUL13910
MUL13920
MUL13930
MUL13940
MUL13950
MUL13960
MUL13970
MUL13980
MUL13990
MUL14000
MUL14010
MUL14020
MUL14030
MUL14040

```



```

EXF (62560B)          ENI (0)          • CLEAR COMM FLAG 1
LDA (CODE1)           STA(ICODE)      • INITIALIZE
ENI 1(1)              ENI 2(0)        •
LDA (ITRKNO)          AJP3(1000)      • CHECK VALUES
AJPO(1000)             INA (-17)       •
AJP3(1CONT)           SLJ(1000)       •
LDA (NUMPTS)          AJP3(2000)      •
AJPO(2000)             SAU (1CHEK)    •
SAU (2CHEK)            INA (-33)       •
AJP3(2CONT)           SLJ(2000)       •
LDA(CHAR)              ALS (6)        •
STA(1CHAR)             ENA(0)        •
LDQ(MASK)              ADL (ITRKN)    • PACK WORDS
ALS (6)                ADL (NUMPTS)  •
ALS (12)               LDO (MASK)     •
ADL (1CHAR)             SLJ(2STR1)   •
1STRT ENA (0)           ENI (0)        •
2STRT ALS (12)          ADL 1(IY)     •
ALS (12)               ADL 1(IY)     •
SSH (1CODE)             SLJ (1STOR)   •
ISK1(N)                SLJ (2SIRT)   •
ALS (24)               ADD (FILL)    •
STA2(BUF)               SLJ (1THRU)   •
1STOR STA2(BUF)         INI 2(1)      •
2CHEK ISK1(N)           SLJ (1SIRT)   •
1THRU ENA (17)          ENO (6)      •
SLJ4 (2+26B)            ENI (0)        •
EXF (62560B)           SLJ (IREST)  • CLEAR COMM FLAG 1
1CHAR ZRO(0)            ZRO (0)        •
ICODE ZRO(C)            ZRO (0)        •
1000 PRINT 120          •
120 FORMAT (//20H TRACK NO. IN ERROR,/) •
SLJ (IREST)            ZRO (0)        •
2000 PRINT 130          •
130 FORMAT (// 23H NUMBER OF PTS IN ERROR //) •
SLJ (IREST)            ZRO(0)        •

```

```

1REST ENI1(N)          • RESTORE
2REST ENI3(N)          • ALL
3REST ENI5(N)          • INDEXES
END

SUBROUTINE DTTRACK LTRKNO, CHAR, NUMPTS, IX, IY
LOC (Z=0, BUF=600)
CON (CODE1=25252525252525B, MASK=00000000000000778,
* MASK1=00000000000000007060)
* CON (FILL=00000000007707778)
DIMENSION IX(8), IY(8)

* SAVE ALL INDEXES
ISAVE SIU1(IREST)    • SAVE
SIU3(2REST)          • ALL
SIU5(3REST)          • INDEXES
ENI (-0)
STA3(BUF)            • SET COMM FLAG 1
EXF (62540B)          • SET 160 CODE WORD
STA (BUF)             • SEND TO SATELLITE
ENQ (6)               • CLEAR COMM FLAG 1
EXF (62560B)          • INITIALIZE
LDA(CODE1)
ENI1(1)
LDA (LTRKNO)          • CHECK VALUES
AJPO(1000)
INA (-9)
SLJ(1000)
AJP3(2000)
SAU (1CHEK)
INA (-9)
SLJ(2000)
ALS (6)
STA(1CHAR)
ENA(0)
ADL (ITRKN0)
LDQ(MASK)
ALS (6)
ALS (12)
ADL (1CHAR)
LDA (NUMPTS)
LDQ (MASK1)
SLJ(2STRT)

```

```

1SIRT ENA (0)          ENI (0)          MUL15130
2SIRT ALS (12)         ADL1(IY).      MUL15160
     ALS (12)         ADL1(IY)      MUL15150
     SSH (1CODE)       SLJ (1STOR)    MUL15160
     ICHEK ISK1(N)     SLJ (2SRT)    MUL15170
     ALS (24)          ADD (FILL)    MUL15160
     STA2(BUF)        SLJ (1THRU)   MUL15170
     STA2(BUF)        INI2(1)     MUL15200
     1STOR ISK1(N)     SLJ (ISIRI)   MUL15210
     2CHEK ISK1(N)     END (6)      MUL15210
     1THRU ENA(17)     ENI (0)      MUL15210
     SLJ4(Z+26B)      ENI (0)      MUL15210
     EXF(625608)      SLJ (IREST)  MUL15210
     1CHAR ZRO(0)      ZRO (0)     MUL15210
     1CODE ZRO(0)      ZRO (0)     MUL15210
1000 PRINT 120          .           MUL15210
120 FORMAT (//20H TRACK NO. IN ERROR,/) MUL15210
     SLJ (IREST)     ZRO (0)     MUL15210
2000 PRINT 130          .           MUL15210
130 FORMAT (//23H NUMBER OF PTS IN ERROR //) MUL15300
     SLJ (IREST)     ZRO(0)     MUL15310
     IREST EN1(N)     ENI2(N)    MUL15320
     2REST EN13(N)    ENI4(N)    MUL15330
     3REST EN15(N)    ENI6(N)    MUL15350
     EXF (425608)    EXF (625608) MUL15360
     END              .           MUL15370
SUBROUTINE DSTATUS (ITYPE, NWIND, IW, INAME, IX, IY)
LOC(IZ=0, BUF=600)    .           MUL15370
CON(SCODE=000300000000007C6B, MASK=0000000000000778,
* MASK1=00000000000000777B, MASK2=000000000000001B,
* MASK3=00000000000000007B, MASK4=0000777777777778)

* SAVE ALL INDEXES
SAVE SIU1(IREST)    SIL2(IREST)  MUL15450
SIU3(2REST)          SIL4(2REST)  MUL15460
SIU5(3REST)          SIL6(3REST)  MUL15470
EXF (625608)        LDA (SCODE)  MUL15480
                                         .SET COMM FLAG 1

```

```

MUL154700
MUL155000
MUL155100
MUL155200
MUL155300
MUL155400
MUL155500
MUL155600
MUL155700
MUL155800
MUL155900
MUL156000
MUL156100
MUL156200
MUL156300
MUL156400
MUL156500
MUL156600
MUL156700
MUL156800
MUL156900
MUL157000
MUL157100
MUL157200
MUL157300
MUL157400
MUL157500
MUL157600
MUL157700
MUL157800
MUL157900
MUL158000
MUL158100
MUL158200
MUL158300
MUL158400

STA (BUF) ENA (1) SET 160 CODE WORD
ENO (6) SLJ4 (Z+2668) •SEND TO SATELLITE
EXF (62560B) ENI (0) •CLEAR COMM FLAG 1
ENA (77777B)
ENA (17) IJP3(L)
ENA (0) AJP3(1000)
LDA (NWIND) INA (-17)
AJP0(1000) SLJ (1000)
ENO (0) ENQ (0)
LRS (12) STA (1ADD1)
STO(2ADD1) LDA ((NAME))
ENO (0) LRS (12) SHIFT CONTENTS
STA (1CONT) STQ (2CONT) OF ADDRESS
ENA (0) LDQ (MASK)
ADL (NWIND) ALS (6)
ADL (IW) ALS (12)
LDO (MASK1) ADL (IX)
ALS (12) ADL (IY)
ALS (1) LDQ (MASK2)
ADL (1TYPE) ALS (11)
LDO (MASK3) ADL (1ADD1)
STA (BUF) LDA (2ADD1)
LDO (MASK4) ADL (1CONT)
STA (BUF+1) LDA (2CONT)
STA (BUF+2) ENI (0)
ENA (17) ENQ (6)
SLJ4 (Z+2668) ENI (0)
EXF (62560B) SLJ (1REST)
IADD ZRO (0) ZRO (0)
2ADD ZRO (0) ZRO (0)
ICONT ZRO (0) ZRO (0)
2CONT ZRO (0) ZRO (0)
PRINT 100
100 FORMAT("// 21H WINDOW NO. IN ERROR ")
SLJ (1REST) ZRO (0)
ENI1(N) ENI2(N) •RESTORE
1REST •ALL

```

```

2REST ENI3(N)          • INDEXES
3REST ENI5(N)          •
END
    MACHINE PARAMS (1A,2A,3A,4A,5A,6A,7A,8A)
    LOC (Z=0, BUF=600)
    CON (SCODE = 000000000000707B)

* 1STAR SLJ (N)          • EXIT/ENTRY
    1A   ZRO (0)          •
    2A   ZRO (0)          •
    3A   ZRO (0)          •
    4A   ZRO (0)          •
    5A   ZRO (0)          •
    6A   ZRO (0)          •
    7A   ZRO (0)          •
    8A   ZRO(0)           •
    ISAVE SIU1(1REST)     •
    SIU3(2REST)          •
    SIU5(3REST)          •
    EXF (62540B)         • SET COMM FLAG 1
    STA (BUF)            • SET 160 CODE WD.
    ENO (6)              • SEND TO SATELLITE
    EXF (62560B)         • CLEAR COMM FLAG 1
    ENA (8)              • READ FROM 160
    SLJ4(2+26B)          •
    ENI1(0)              •
    SAU (1CHEK)          • TRANSFER BUFFER
    ITRAN LDA1(1A)        • TO ARG5
    LDA1(BUF,)           •
    ARS (36)             •
    ICHEK ISK1(N)         • RESTORE INDEXES
    1REST ENI1(N)          •
    2REST ENI3(N)          •
    3REST ENI5(N)          •
    SLJ (1STAR)           •
    END

MUL1585C
MUL15860
MUL15870
MUL15880
MUL15890
MUL15900
MUL15910
MUL15920
MUL15930
MUL15940
MUL15950
MUL15960
MUL15970
MUL15980
MUL15990
MUL16000
MUL16010
MUL16020
MUL16030
MUL16040
MUL16050
MUL16060
MUL16070
MUL16080
MUL16090
MUL16100
MUL16110
MUL16120
MUL16130
MUL16140
MUL16150
MUL16160
MUL16170
MUL16180
MUL16190
MUL16200

```

**UNCLASSIFIED**

Security Classification

**DOCUMENT CONTROL DATA - R&D**

(Document classification or title, body of abstract and indexing information must be entered when the report is classified)

1. ORIGINATING ACTIVITY (Corporate name)		2a. REPORT SECURITY CLASSIFICATION
UNITED STATES NAVAL POSTGRADUATE SCHOOL MONTEREY, CALIFORNIA		UNCLASSIFIED
3. REPORT TITLE		3b. GROUP
AN ON-LINE SIMULATION OF ASW IN A MULTI-BURST NUCLEAR ENVIRONMENT		
4. DESCRIPTIVE NOTES (Type of report and analysis code if any) Master's Thesis		
5. AUTHOR(S) (Last name, first name, middle) McMichael, David L.		
6. REPORT DATE May 1966	7a. TOTAL NO. OF PAGES 123	7b. NO. OF REPS 7
8a. CONTRACT OR GRANT NO.	8b. ORIGINATOR'S REPORT NUMBER(S)	
9a. PROJECT NO.	9b. OTHER REPORT NUMBER(S) (Any other numbers that may be assigned this report)	
10. AVAILABILITY/LIMITATION NOTICES  Qualified requesters may obtain copies of this report from DDC.		
11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY U. S. Naval Postgraduate School Monterey, California	
13. ABSTRACT  A general approach is documented as a guide to aid in the formulation and implementation of on-line, real time computer simulations. A computer program MULNUCL, is developed as an on-line, real time computer simulation of antisubmarine warfare in a multiple burst nuclear environment. The principals of the game are a submarine armed with torpedoes, and two destroyers equipped with stand-off antisubmarine weapons. The simulation is intended as a demonstration of the on-line capabilities of the United States Naval Postgraduate School computer system and as a tool for further study of the factors involved in a representative ASW operational environment.		
<b>Reproduced From Best Available Copy</b>		

## UNCLASSIFIED

## Security Classification

14 KEY WORDS	LINE A		LINE B		LINE C	
	ROLE	WT	ROLE	WT	ROLE	WT
Simulation						
War Game						
ASW						
Multi-burst Nuclear Environment						
<b>INSTRUCTIONS</b>						
1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization (corporate author) issuing the report.	imposed by security classification, using standard statements such as:					
2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.	(1) "Qualified requesters may obtain copies of this report from DDC."					
2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, where applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.	(2) "Foreign government and dissemination of this report by DDC is not authorized."					
3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parentheses immediately following the title.	(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."					
4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.	(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."					
5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.	(5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."					
6. REPORT DATE: Enter the date of the report as day, month, year. If more than one date appears on the report, use date of publication.	If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.					
7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.	11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.					
7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.	12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring (paying for) the research and development. Include address.					
8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.	13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.					
8b, 8c & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, sub-project number, system number, task number, etc.	It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as (TS), (C), or (U).					
9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.	There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.					
9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers (either by the originator or by the sponsor), also enter this number(s).	14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, code name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.					
10. AVAILABILITY/LIMITATION NOTICE: Enter any limitations on further dissemination of the report, other than those						

DD FORM 1 JAN 64 1473 (BACK)

124

UNCLASSIFIED

Security Classification